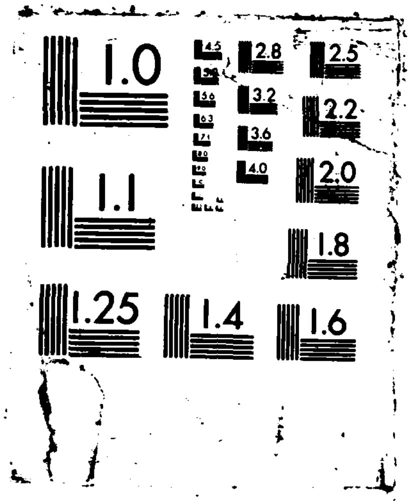
AD-A189 511 A NETWORK APPROACH TO RATED OFFICER GATE MANAGEMENT(U) 1/3
AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL
OF ENGINEERING M S OLSON DEC 87 AFIT/GOR/ENS/87D-13

UNCLASSIFIED F/G 5/9 NL


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0

AFIT/GOR/ENS/87D-13

A NETWORK APPROACH TO RATED OFFICER GATE MANAGEMENT

THESIS

Mark S. Olson
Major, USAF

AFIT/GOR/ENS/87D-13

Approved for public release; distribution unlimited

AFIT/GOR/ENS/87D-13

A NETWORK APPROACH TO RATED OFFICER GATE MANAGEMENT

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Operations Research

Mark S. Olson

Major, USAF

December 1987

Approved for public release; distribution unlimited

Preface

As is probably true for many research efforts, the initial goals of this project far exceeded what has been achieved. This project has impressed upon me the immense challenge that faces those whose charter it is to bring automation to large systems such as those encountered in military personnel management.

I thank Major Brian Sutter for the extensive long distance support that he provided from the Analysis Division at the Air Force Military Personnel Center. His technical backing made this project possible.

My advisors, Lieutenant Colonel Bill Rowell and Major Joe Litko, provided the guidance necessary to permit a fledgling analyst to complete a relatively large project such as this thesis effort. My thanks to them.

As always, my wife Terri and our children Victoria and Michael did more than their part in helping me to succeed. Their love and support often go unthanked, but never go unappreciated.

Mark Olson

Table of Contents

Page

Preface....................................ii

List of Figures............................vi

List of Tables.............................vii

Abstract...................................viii

I. Introduction............................1

The Issue..................................1
The Research Problem.......................2
Problem Statement..........................2
The Research Question......................2
Research Objectives........................3
Scope of the Study.........................3
Terminology................................4

II. Background and Literature Review.......8

Rated Gate Management......................8
Present Implementation.....................10
Personnel Modeling Approaches..............14
Descriptive Methods........................14
Entity Flow Simulations....................15
System Dynamics Simulations................16
Prescriptive Methods.......................17
Probabilistic Models.......................17
Linear Programming Methods.................18
Network Flow Methods.......................19
Summary....................................21

III. Methodology...........................23

Methodology Considerations.................23
Information Requirements...................23
Available Data.............................25
Key Relationships..........................26
Methodology Selection......................27
Conceptual Modeling Approach...............27
Computerized Network Model.................32
Modeling Routine...........................32
Network Solution Algorithm.................33
Model Assumptions..........................35
Model Structure............................36
Model Objective Function...................37
Network Node Parameters....................38

Time Periods....39
Duty Types....41
Aviation Service Date Year Groups....43
Flying Gate Credit Accumulation....44
Network Dimensions....45
Network Arcs....46
External Flows....47
Rotations and Gains....47
Attrition Losses....48
End Requirements....50
Model Constraints....50
Structural Constraints....50
Side Constraints....52
Model Inputs....55
Input Data File....56
Input Parameters....56
Model Outputs....58
Network Data Files....58
Information Output File....59
Auxiliary Output Files....59
Screen Output....59
Model Run Times....59
Model Verification....60
Structure-Verification Test....61
Boundary Adequacy Test....62
Model Validation....63
Behavior Reproduction....64
Behavior Sensitivity....65
Summary....65

IV. Results....67

Solution Considerations....67
Multiple Optimality....67
Infeasibility....69
Sensitivity....72
Solution Analysis....72
Duty Durations....73
Gate Failure Costs....78
Experience Requirements....81
Manning Requirements....82
Attrition....83
Initial State of the System....84
Practical Implications....85

V. Observations and Recommendations....86

Adaptability of the Model....86
Shortcomings and Limitations....88
Level of Detail....88

Attrition....89
Grouped Rotations....91
Duty Durations....91
Measuring Gate Attainment....92
Information Format....93
Input Data Collection....94
Recommendations....94
Refinement....94
Revisions....95
Conclusion....97

Appendix A: GATES User's Guide....98

Appendix B: GATES Program Listing....106

Appendix C: Samples of Network Data Files....161

Appendix D: Sample of NETSID Output....162

Appendix E: GATES Output File SANITY.OUT....163

Appendix F: GATES Output File ROTEPLAN.OUT....173

Appendix G: GATES Output File NODEARC.OUT....197

Appendix H: Sample of GATES Screen Output....198

Appendix I: Input/Output Data Summary....199

Bibliography....218

Vita....222

List of Figures

Figure | Page

1. Example Output From Current Arithmetic Model.......13
2. Two-Dimensional Assignment Network................29
3. Three-Dimensional Assignment Network..............30

List of Tables

Table | Page

1. Tour Length vs. Rotation/Assignment Distribution....75
2. Tour Length vs. Gates Missed........................76
3. Gate Costs vs. GATES Model Results..................79
4. Manning Tolerance vs. Assignment Distribution.......83

AFIT/GOR/ENS/87D-13

Abstract

The purpose of this study was to provide Air Force rated officer managers at the Air Force Military Personnel Center with a decision aid for the management of rated officer flying gates. Air Force rated officers are those officers who hold an aeronautical rating and are authorized to perform duties as pilots or navigators. Flying gates are milestones that must be achieved at certain phase points of a rated officer's career.

This study resulted in development of a single commodity network flow model with side constraints. This model is designed to represent the rotation of rated officers between flying and nonflying duties and provides a means for measuring overall attainment of flying gates. It is an aggregate model which provides general assignment guidance aimed at minimizing nonachievement of flying gate requirements, while maintaining required manning levels in flying and nonflying duties.

Initial analysis of model outputs indicates that the model solution may provide an avenue to improved gate management. Shortcomings of the model that bear further study include the level of detail provided by the model and the method used to model attrition of the rated officer force.

A NETWORK APPROACH TO RATED OFFICER GATE MANAGEMENT

I. Introduction

The Issue

The personnel resource managers at the Air Force Military Personnel Center (AFMPC) are responsible for a myriad of tasks related to personnel management. These tasks include assignment of rated officers--aircraft pilots and navigators--to flying and nonflying duties. The Aviation Career Incentive Act of 1974 dictates that rated officers must accumulate specified numbers of active flying months at various phase points in their careers--these phase points are often referred to as "gates"--to be eligible to receive Aviation Career Incentive Pay (8:12). Air Force policy dictates that rated officer careers will be planned to insure attainment of these gates (8:12). However, the Air Force employs numerous rated officers in positions not involving flight duties. The rotation of rated officers into and out of these nonflying positions requires close management to insure compliance with gate requirements.

As an aid to personnel managers, the Analysis Division at AFMPC currently uses a simple computational (arithmetic) model to identify potential problems in rated gate management (32). The problem with the existing method is that it does not consider the dynamics of the rated officer force. The existing model takes a snapshot look at the current rated gate situation. The model cannot identify potential problems that may result due to variations in rated force allocations, changes in the number of authorized cockpit positions, fluctuations in the number of personnel that are available for reassignment, or changes in attrition rates (32).

Due to these shortcomings with the existing gate management model, the Analysis Division at AFMPC has identified the need for a model which can incorporate the dynamics of the Air Force rated officer force (32). The primary expected benefit of such a model is improved management of rated officer assignments.

The Research Problem

Problem Statement. The existing computational model used by AFMPC for rated officer gate management fails to encompass the dynamic aspects of rated officer career movements. As a result, effective management of the rated officer force is hampered.

The Research Question. The overall research question that this study addresses is as follows:

> How can rated officer career movements be modeled to provide AFMPC personnel managers the information necessary for effective management of the rated officer force?

Research Objectives. The overall objective of this study is to provide rated officer force managers at AFMPC with a means of identifying potential rated management problems. To accomplish this overall objective, this study addresses the following sub-objectives:

> 1. Determine the specific information required by AFMPC rated officer force managers;
>
> 2. Determine what input data is readily available from AFMPC databases;
>
> 3. Determine the key interrelationships that affect the rated officer force structure;
>
> 4. Determine which modeling methodologies can be applied to this problem;
>
> 5. Determine an output format (method of information presentation) that depicts the information needed by AFMPC managers.

Scope of the Study

The scope of this research is limited to rated officer manning problems that are within the domain of officer personnel resource managers at AFMPC. This study does not attempt to project external factors that impact the rated officer management problem, such as officer retention rates, manning authorizations, and weapon system inventories. However, these factors--as projected by other methods--are used as model inputs.

This study takes an "aggregate" approach to the rated officer gate issue. No attempt is made to identify those individuals who may experience problems with rated gate compliance. Rather, the goal is to identify those groups of rated officers that require close management to insure gate compliance.

Terminology

The following are some terms used in this study that may not be familiar, or which have a specific (unfamiliar) meaning in the context of this study.

Advanced Student. Advanced students are those rated officers who are undergoing some type of advanced flight training. Advanced students have already earned their "wings" (aeronautical ratings). (9:page 3-1)

AFIT. This category of duty assignments includes graduate degree programs at the Air Force Institute of Technology or at civilian institutions. Assignments are generally 15 to 18 months in duration and result in a requirement to serve in a nonflying duty for a minimum of three years following graduation (8:45).

ASTRA. "Air Staff Training". This is a one-year nonflying duty assignment available to a select few junior officers (approximately six to seven years commissioned service).

Attrition Rate. This is a percentage or proportion of individuals belonging to a specified group that separate from the Air Force within a specified time period, usually one year. Normally, attrition rates for rated officers are calculated based on aviation service date year groups.

Aviation Career Incentive Pay. This is the formal name for "flight pay." Its basic purpose is to retain sufficient numbers of rated officers to meet Air Force rated manning requirements.

Aviation Service Date (ASD). The actual aviation service date is the date an individual began flying duties with the Air Force. In the context used in this study, aviation service date refers to the number of years (or portions thereof) since initial aviation service date.

Continuation Rate. This is the percentage or proportion of individuals belonging to a specified group that continues in military (Air Force) service. The rate applies to a specific time period, usually one year. If the attrition rate is .12 (or 12 percent) then the continuation rate is 1.00 - .12 = .88 (or 100 - 12 = 88 percent). The cumulative continuation rate over a given number of time periods is the product of the respective continuation rates for the individual time periods.

Experience Level. Within each major weapon system (aircraft) group, regulations specify the minimum number of flying hours and/or years of aviation duty required for an individual to be classified as "experienced." The overall experience level within a flying organization is determined by the percentage of members of the organization that are classified as "experienced." Organizational experience level is tracked as a management device. (9:6-9 thru 6-26)

First Assignment Instructor Pilot (FAIP). These individuals are assigned to instructor duties in Air Training Command immediately following Undergraduate Pilot Training. After approximately three years of duty as an instructor, most FAIPs are assigned to flying duties in one of the major weapon system groups.

Fiscal Year. This is the period from 1 October through 30 September. This is the time period around which planning and budgeting is focused in the Federal government.

Flying Gate. A flying gate is a milestone or phase point based on the number of years that an individual has performed active flying duties. Three flying gates are specified in Air Force Regulation 36-20 for management of rated officers: the first gate (six-year gate) requires six years of flying within the first 12 years of aviation service; the second gate (nine-year gate) requires nine years of flying within the first 18 years of aviation service; the third gate (11-year gate) requires 11 years of active flying within the first 18 years of aviation service. (8:12)

Major Weapon System Group. This is a category of aircraft (with similar missions) used for management of the rated officer force. Generally, there is very little flow of officers from one major weapon system group to another. There are eight major weapon system groups: tactical fighter/reconnaissance, bomber, tanker, strategic airlift, tactical airlift, helicopter, trainer, and "mission". (9: 3-4 thru 3-9; 33)

Professional Military Education (PME). Though it takes several forms, the PME of interest in this study is that which is accomplished as a full time (nonflying) duty assignment of just less than a year in duration. The two categories of PME of importance in this study are Intermediate Service School (ISS), attended by majors, and Senior Service School (SSS), attended by lieutenant colonels and colonels.

Rated Officer. This is a general term used to refer to Air Force officers possessing an aeronautical rating (pilot or navigator) (7:11).

Rated Staff. In general, this refers to staff duties performed by rated officers--some of which also involve flying duties. For the purpose of this study, rated staff refers only to nonflying staff duties.

Rated Supplement. This refers to duties performed by rated officers that are traditionally performed by nonrated officers. These are nonflying duties. (8:36)

Undergraduate Flying Training (UFT). This term refers to the flying training that an individual must accomplish to receive an aeronautical rating ("wings"). UFT includes Undergraduate Pilot Training (UPT), Undergraduate Navigator Training (UNT), and Undergraduate Helicopter Training (UHT).

II. Background and Literature Review

The literature reviewed in this section provides background information on the rated officer management issue and provides brief descriptions of various personnel models and modeling methodologies. Most of the material supporting the rated management background discussion is from Air Force documents. Most of the articles related to modeling methodologies were collected from technical journals and the Defense Logistic Agency's Defense Technical Information Center computerized database.

The literature is reviewed in a topical order, beginning with a discussion of the impetus behind rated gate management. The background discussion is followed by a brief look at some of the personnel models and modeling methodologies currently used in personnel management. The discussion focuses on the applicability of these models and methodologies to the rated gate management issue.

Rated Gate Management.

The Aviation Career Incentive Act of 1974 established specific requirements for the management of rated officers (25:2; 8:12). This act increased the Air Force's flexibility in assigning rated officers to nonflying duties, while still insuring the officers' eligibility to receive Aviation Career Incentive Pay--commonly referred to as "flight pay" (25:2). The Air Force considers flight pay to be a major factor in retention of rated officers (6:2) and has established policies which help assure rated officers that they will be entitled to flight pay for all or at least most of their careers (8:12). This assurance also increases the willingness of rated officers to occupy those nonflying positions which the Air Force believes should be filled with officers having rated aviation experience. This "rated supplement" and "rated staff" force, as it is called, provides a pool of rated officers that are readily available to fill cockpit positions under emergency conditions (8:36). This supplement/staff force also provides what is known as "rated presence" and "rated expertise" in career areas not directly involved in active flying operations. This presence is important because nearly all Air Force activities have some impact on present or future flight operations.

Air Force Regulation (AFR) 36-20 implements the Aviation Career Incentive Act and delineates the requirements for entitlement to continuous receipt of flight pay when assigned to nonflying duties. These requirements are as follows:

> 1. Perform six years of operational flying by the 12th year of aviation service;
>
> 2. Perform nine years of operational flying by the 18th year of aviation service;
>
> 3. Perform 11 years of operational flying by the 18th year of aviation service to receive flight pay through 25 years of officer service (8:12).

Rated officers assigned to flying duties are entitled to receipt of flight pay--regardless of whether they have achieved these milestones.

The management policies which help insure maximum achievement of these rated officer utilization standards--known as "gates"--are also stated in AFR 36-20. "It is the Air Force policy that as many members as possible perform at least 9 years of operational flying duty during the first 18 years of aviation service" (8:12). The regulation further states that graduates of pilot training and navigator training "are assigned to operational flying duties until they have completed at least 6 years of operational flying duties" (8:12). More experienced rated officers are also assured of being able to meet their gates:

> The typical officer with over 12 years' aviation service must have completed or be able to complete at least 9 years, and preferably 11 years, of operational flying duty before the 18th year of aviation service before being assigned to nonoperational flying duty [8:12].

Compliance with these policies requires careful management of the rated officer force (32).

Present Implementation. The AFMPC personnel resource managers are charged with matching Air Force personnel to manpower positions. These "assignment officers," as they are sometimes called, assign individuals to duty positions when vacancies occur (due to resignations and retirements, completion of training, rotations from overseas, and so on). Various rules are used to determine which individuals are to be matched to which positions. The rated officer management policies stipulated in AFR 36-20 are among the important decision factors in the assignment process.

To keep the assignment officers informed of large scale and long term trends in the rated officer force, AFMPC analysts conduct periodic analyses and brief their findings to the personnel managers (31). The analysis applicable to the rated gate management issue involves use of a simple computational model. This method involves computation of the "gate supportable inventory", which is "the maximum personnel inventory that will allow all members to complete a given number of flying gates" (17:1). A rudimentary comparison is then made to the actual current personnel inventory. Separate computations and comparisons are made for each major weapons system (aircraft) group and each aeronautical rating. For each of these aircraft groups and ratings, general assessments are made regarding the capability to comply with flying gate policies (18).

The major strength of the current methodology is its simplicity. All data required in the computations is readily available from the AFMPC personnel database. The computations are straight-forward and the results are easy to understand. However, there appear to be a few major shortcomings of this method. (32)

First, some of the assumptions underlying the computation of the gate supportable inventory appear to be unrealistic. One assumption made is that only rated officers still requiring gate credit occupy flying positions (17:1).

Another assumption is that "there are no management or assignment constraints" (17:1). Real world considerations such as experience level requirements and assignment/rotation policies are contrary to these assumptions.

Recognizing that some of the assumptions are not fully valid, the method employs a measure called "management flex". Management "flex" is "the difference between the current [one through] 18 [year] aviation service population and the gate supportable inventory" (17:2). The problem is that the actual management "flex" required by real world constraints is unknown. However, the AFMPC Analysis Division considers the minimum practical "flex" required to be equal to about 15 percent of the gate supportable inventory (17:2). Actually, the amount of "flex" required probably varies with time and from one major weapon system group to another.

Figure 1 shows an example of one type of output produced by the arithmetic model (18). The graph in the figure resulted from a recent analysis of the strategic airlift pilot force. The wide gap between the gate supportable inventory and the current inventory of pilots in the one through 18 year aviation service groups--the management "flex"--seems to indicate that there should be very little difficulty in complying with flying gate requirements. However, there have in fact been some difficulties in meeting gate requirements for this group of rated officers (31).

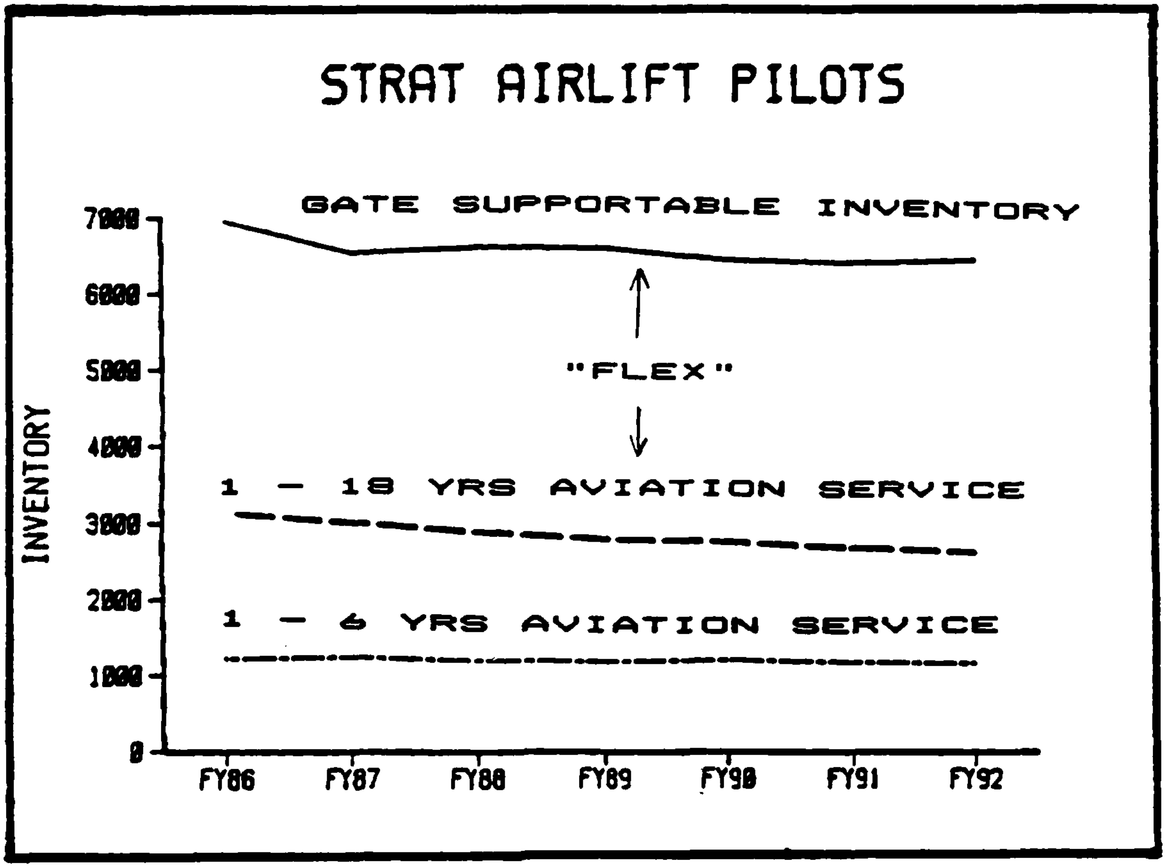


Figure 1. Example Output From Current Arithmetic Model (Adapted from 18)

This example illustrates the most significant shortcoming of the current model. That is, the model fails to adequately identify potential gate management problems. As shown by the relatively smooth lines on the graph in Figure 1, the current method provides no means of identifying the effects of fluctuations in the rated officer force structure over time. Assignment officers may be able to deal more effectively with these fluctuations if they have sufficient notice. (31)

Personnel Modeling Approaches

The remainder of this literature review examines possible methodologies for an improved gate management model. The personnel models most commonly encountered in the literature fall into the following categories:

1. Descriptive methods.
   a. Entity flow simulations.
   b. System dynamics simulations.
2. Prescriptive (optimization) methods.
   a. Probabilistic models.
   b. Mathematical programming.

Existing methodologies in each of these categories are discussed and their application to the gate management problem is considered.

Descriptive Methods. Descriptive models are designed to describe the underlying systems. In general, a descriptive model does not directly provide an answer or solution--though it can give insight into possible solutions through repeated experimentation with the model. A descriptive model can be as simple as a graphical representation of the real world system. A type of descriptive model frequently used in business is the spreadsheet model. However, the various forms of computer simulation are the most commonly employed descriptive methods for modeling complex systems such as large personnel systems.

Simulation is a method of arriving at an approximate solution to a problem that cannot be stated in precise mathematical terms or has no known mathematical solution. Simulation models can also provide some insight into cause-and-effect relationships within real world systems. Simulation models can be categorized into two general groups: entity flow simulations and system dynamics simulations.

Entity Flow Simulations. Entity flow simulations treat each object or unit within the system individually. In a manpower system simulation, each person would be represented in the model. Charpie (3) effectively employed such a simulation model in studying problems with the B-52 navigator force. However, one of the problems with a model of this type is that a large system requires a large amount of data for input into the model and the resulting simulation requires long computer run times (4:42).

Existing entity flow simulations generally lack adequate feedback loops to make realistic adjustments for changing conditions during a simulation run (23). For example, if a given set of inputs resulted in assignment (by the model) of all rated officers to nonflying positions (possibly due to limitations with the particular model used), the model outputs would not provide insight into what might really occur under the specified set of conditions. In actuality, senior Air Force managers would probably recognize such a problem and make corrections to avoid such a result.

System Dynamics Simulations. Recognizing some of the limitations of entity flow simulation models, Forrester devised a methodology called "system dynamics" (12). The real advantage of a system dynamics approach is that this approach incorporates feedback loops so as to more realistically represent real world systems interactions (12:12-14). One shortcoming of system dynamics models is that they do not represent systems down to the minute levels of detail that are possible with entity flow simulations.

System dynamics models are frequently used to examine the consequences of changes in policy (12:8). Clark and Lawson provide a system dynamics simulation model to evaluate policy effects on a particular segment of the Air Force enlisted personnel force that is subjected to a high number of overseas assignments (4). Also, Knight employs system dynamics modeling to examine the impact of various factors such as retention and force authorizations on the allocation of Air Force pilots (22). A system dynamics approach might readily be applied to the rated gate management problem. Such an approach would reduce the input data requirements and may provide better insight into the future effects of changes in policy and other factors such as retention or personnel authorizations. However, the lack of detail inherent in this approach may result in insufficient information to assist AFMPC assignment personnel in managing fluctuations in the rated force.

Prescriptive Methods. The general purpose of an optimization model is to find an optimal or "best" solution for a given set of inputs. In other words, an optimization model prescribes a solution. Optimization models come in several forms and can be classified in many ways. Two broad classifications of optimization models and techniques are probabilistic modeling (which includes Markov processes) and mathematical programming (14). The types of mathematical programming most frequently encountered in the literature on personnel modeling are the various forms of linear programming (including the goal programming problem, the transportation problem, the transshipment problem, and the assignment problem) and network flow programming. There is much overlap between these methods--for instance, a network model may be formulated as a linear program (14:319).

Probabilistic Models. Several researchers have discussed the application of Markov solution methods to manpower models (5; 26). Methods have been developed to solve for the steady state of manpower systems in which personnel are divided into ranks or grades with known probabilities of transition from one grade to the next over discrete or continuous time intervals (26:249). In the context of rated gate management, the ranks could represent degree of gate attainment. However, the transition probabilities from one level of gate credit to the next would generally not be known--though they could possibly be estimated.

Linear Programming Methods. Numerous linear programming techniques have been used to model personnel systems. The basic method of solving linear programming problems is the simplex method developed by Dantzig in 1947 (14:52). Since then numerous modifications to this method have been developed.

Aronson, Morton, and Thompson developed the "forward simplex method," which is a modification of the basic simplex method for solving linear programs with multiple time periods (1). Aronson and Thompson (2) subsequently applied the forward simplex method to solve personnel planning problems. The manpower planning problem they addressed involves several grades of personnel, forecasted personnel goals for each grade, and transition probabilities between grades that are described by a Markov probability matrix. Constraints on the system include budgets for each time period (applied to personnel salaries) and limits on the total number of personnel. They report considerable improvement in computer solution time over the basic simplex method. Modifications to their method may be applicable to the rated gate problem, but the Markov transition probabilities are generally unknown (though perhaps estimable) for the gate problem.

Linear goal programming is one of the many adaptations and extensions of linear programming. Goal programming permits the combination of several goals into a single objective function (37:358).

Siverd and Thompson (28) describe a modified goal programming approach--they call it "ratio goal programming"--that attempts to maximize organizational effectiveness by attainment of specific personnel assignment ratios. In other words, given that a preferred mix of personnel skills or experience levels can be established, their approach is to determine the correct levels of the factors (such as salaries or recruiting expenditures) that will result in the desired mixes or ratios (28). This method has potential for application to gate management if, for instance, the preferred proportion of rated officers attaining their 11 year gate can be established.

Network Flow Methods. Network flow theory has much overlap with linear programming theory. However, the specialized structure of the network problem sometimes permits application of solution algorithms that are more efficient than the traditional simplex method. In addition, the network structure sometimes provides a useful means for conceptualizing the model for a large system.

A network can be thought of as a series of locations (nodes) connected by paths or routes (branches or arcs) (14:297). A network model of a manpower system could treat the organizations or duties to which people are assigned as nodes and could treat the possible rotations, promotions, or transfers of personnel as the arcs. This has potential application to the gate management model, since flying and nonflying duties could be treated as nodes and the assignment decisions could be treated as arcs.

Various network programming algorithms have been applied to a variety of personnel systems. Thompson (36) proposes the use of a network transshipment model to approximate a linear program of an extremely large manpower system; such an approximation can result in considerable decreases in computer run time--perhaps without excessive loss of detail.

Klingman, Mead, and Phillips describe the application of network solution techniques to two military manpower planning problems (21). The two prototype models that they describe are an Army enlisted personnel assignment model and an Army officer strength forecasting model. They point out the advantage of network optimization techniques for solving large personnel problems: network techniques "are typically 10 to 100 times faster than linear programming optimizers" (21:787).

Specialized network optimization algorithms have been devised to deal with various special network constructs. Price and Gravel offer a means of solving a network problem that has side constraints (27:196-202) Side constraints are constraints on the solution that are not inherently modeled within the network--that is, constraints other than upper and lower bounds on arc (flow) capacities or the standard network constraints requiring conservation of flow (flow-in equals flow-out) at each node. Such side constraints are readily applicable to manning level requirements, such as those related to duties performed by the rated officer force. Price and Gravel also suggest the application of a heuristic method for dealing with problems containing large numbers of side constraints that can occur in modeling attrition, since increasing the number of side constraints decreases the efficiency advantages of the network algorithms compared with normal simplex methods (27:201-202).

Liang and Buclatin (24) employed a network formulation with side constraints to solve a Navy enlisted personnel assignment problem. The problem solved involved matching 200 people to 230 available jobs at minimum training cost, subject to a limit on the number of training positions available for each of 16 different training courses. The problem involved 4,592 arcs and 16 side constraints. Liang and Buclatin solved the problem using a network computer code called NETSID because of its efficiency over other available packages that can handle networks with side constraints (24:6-8). Kennington and Helgason provide a technical discussion of the algorithm underlying the NETSID code (19:166-174).

Summary

This chapter has provided a detailed discussion of the rated gate management problem, providing some insight into the structure of the real world system. Additionally, several techniques employed in modeling other personnel systems have been examined. These numerous methods provide a broad base for selection of a means of addressing the gate management issue. Additional considerations in selection of a methodology and a detailed description of the method implemented are provided in Chapter 3.

III. Methodology

This section details the methodology employed and the model developed in this study. The section begins with a discussion of the main considerations that influenced selection of a modeling approach. That discussion is followed by a description of a conceptual model that demonstrates the structure underlying the rated gate management issue. The computerized model of the gate management system, as implemented for this study, is then described in detail. Finally, the methods used to verify and validate the model are examined.

Methodology Considerations

Prior to selection of a methodology for application to the rated gate management problem, some important considerations must be addressed:

> 1. What information is required by AFMPC rated officer force managers relative to the gate management issue?
>
> 2. What data is readily available for input to a gate management model?
>
> 3. What are the key interrelationships that describe the rated officer rotation system?

Only after examining these modeling considerations can a methodology be selected.

Information Requirements. One of the primary considerations in selecting a methodology for solving an existing problem is to determine--to the extent practical--what general form the final solution should take. In other words, the type of answer desired should have some influence on the selection of a solution method.

Ultimately, rated force managers may find great utility in a model which produces optimal assignment "decisions" for each individual rated officer--subject to the many rules, policies, and considerations that affect the actual assignment process. Development of such a large scale model must build upon previous work at modeling the rated officer force. A model which provides aggregate information on rated force rotations and their relationship with flying gate requirements and policies could provide a basis for development of such a large scale assignment model.

The arithmetic gate management model currently employed by the AFMPC Analysis Division seems to provide insufficient detail to serve its purpose well. The Analysis Division has indicated the need for more detailed information about projected gate requirements (32). Rated force managers need to know about upcoming bottlenecks in rated officer manning and flying gate attainment, specifically those time periods and rated officer groups requiring careful management to insure compliance with stated policies and requirements (35).

To identify bottlenecks, a descriptive methodology such as simulation may be adequate. However, an optimization approach can serve to describe the system while also prescribing a rated officer assignment policy which maximizes gate attainment subject to manning requirements. A network formulation provides such a capability.

The information requirements also affect the level of detail that must be used in modeling the real world system. A review of rated force management documents indicates that the rated management issue is a relatively large and complex problem (9; 10). For example, the total Fiscal Year 1988 rated officer manning requirement exceeds 32,000 personnel (9:page 3-3). This total requirement is divided between eight manning accounts (duty types such as operational flying, staff, rated supplement, and professional military education), nine major weapon system groups, and three aeronautical ratings (pilots, navigators, and electronic warfare officers) (9:page 3-10, pages 5-6 thru 5-27).

Considering the size of the problem, a solution method that allows some degree of aggregation seems appropriate. Yet, an increase in detail over the current arithmetic model seems essential in order to provide the information necessary for effective management of the rated force. Increasing model detail for large problems requires the use of an efficient solution method. Network programming methods have been shown to be relatively efficient in solving large problems.

<u>Available Data</u>. The availability of data can significantly affect the selection of a modeling methodology since the model input data is directly related to the level of detail that is possible. The AFMPC maintains an extensive

database on military personnel. Discussions with personnel analysts indicated the ability to provide extensive data support relative to the gate management issue (31; 34). Availability of data does not seem to be a limiting factor in this study.

Of course, large data requirements translate into greater effort in collecting model inputs and longer computer run times. Again, the size of the real world system suggests consideration of an aggregate model. The degree of aggregation must be controlled so that critical detail is not lost. A network formulation can provide varying degrees of aggregation and therefore provides some modeling flexibility.

Key Relationships. The relationships that describe the real world system being modeled--the structure of the system--play a key role in selection of a modeling methodology. Complex systems are often modeled using simulation because of the flexibility that simulation provides. On the surface, the rated officer assignment system--with its several underlying policies and constraints--appears to be rather complex.

By reducing the degree of precision required from the model, a network representation of the rated officer management system can be developed. The nodes of the network can be thought of as representing the duty assignments and flying gate status (such as aviation service date year group and gate credit accumulated) of the individuals within the system. The arcs of the network can represent the assignment decisions and the effects of those decisions--that is, a transition from one duty and gate status to another duty and gate status.

Methodol gy Selection. Due to the reasons outlined above, the availability of an efficient network optimization routine (NETSID), and the considerable potential that a network approach holds, a network formulation and optimization methodology was selected for this study.

Conceptual Modeling Approach

Rated officer force duty assignment rotations within each aviation rating (pilot or navigator) and each major weapon system group (such as fighter, bomber, or strategic airlift) can be viewed as a four-dimensional network. The following parameters define the dimensions of the network:

1. The time period which is being examined;
2. The particular duty to which an individual is assigned;
3. The aviation service date (ASD) year group;
4. The amount of flying gate credit that has been accumulated.

Each location (node) in the network can be thought of as representing the time (in years) and an individual's status: his duty assignment, his ASD year group, and the amount of flying credit he has accumulated. The paths (arcs) between nodes can be thought of as representing assignment decisions made by AFMPC assignment personnel.

The node at the starting point of an arc represents the individual's status immediately prior to beginning the duty assignment represented by the arc. The node at the end of the arc represents the individual's status at the completion of the duty tour length. While serving in a duty position, the individual can be thought of as traveling along the arc from the beginning node to the ending node. The duty type associated with the node at the end of the arc represents the duty position occupied once the assignment has occurred, whereas the duty type associated with the node at the begining of the arc represents the individual's duty position prior to the new assignment.

Examining only the time and duty type dimensions, the two-dimensional diagram in Figure 2 illustrates the network relationships. Assume that assignments to duty types A and C have a normal duration of one time period, and assignments to duty type B are normally two time periods in duration. For individuals in duty type A that are available for reassignment at time 1 (node "A1"), the paths labelled "arc 1", "arc 2", and "arc 3" represent the possible duty assignments:

1. Arc 1 represents a new one-year assignment to duty type A.

2. Arc 2 represents a two-year assignment to duty type B.

3. Arc 3 represents a one-year assignment to duty type C.

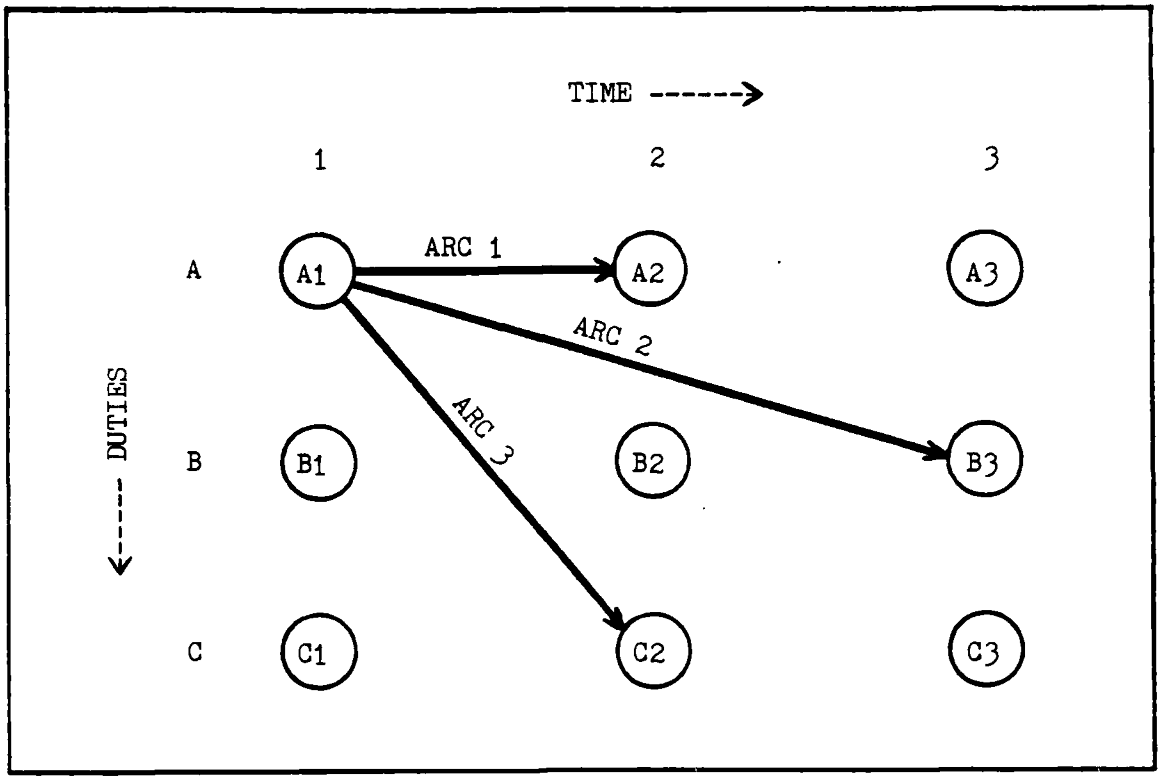


Figure 2. Two-Dimensional Assignment Network

Subject to the duty duration assumptions, these are the only possible assignment paths from node "A1" to duty types A, B, and C. For example, no path exists from node "A1" to node "A3" because such an assignment is not possible if all assignments to duty type A are assumed to have a duration of only one year.

By treating assignments in this manner, arcs terminate at nodes that have a time value corresponding to the time when (on average) individuals will be available for reassignment to new duties. Completion of a duty tour length can be thought of as an "arrival" at the corresponding arc ending node. Such an arrival indicates availability for reassignment and the need for an assignment decision.

Adding a third dimension to the network for flying gate credit accumulation results in a network pattern such as that shown in Figure 3, where two gate credit values are represented by roman numerals I and II. The third dimension

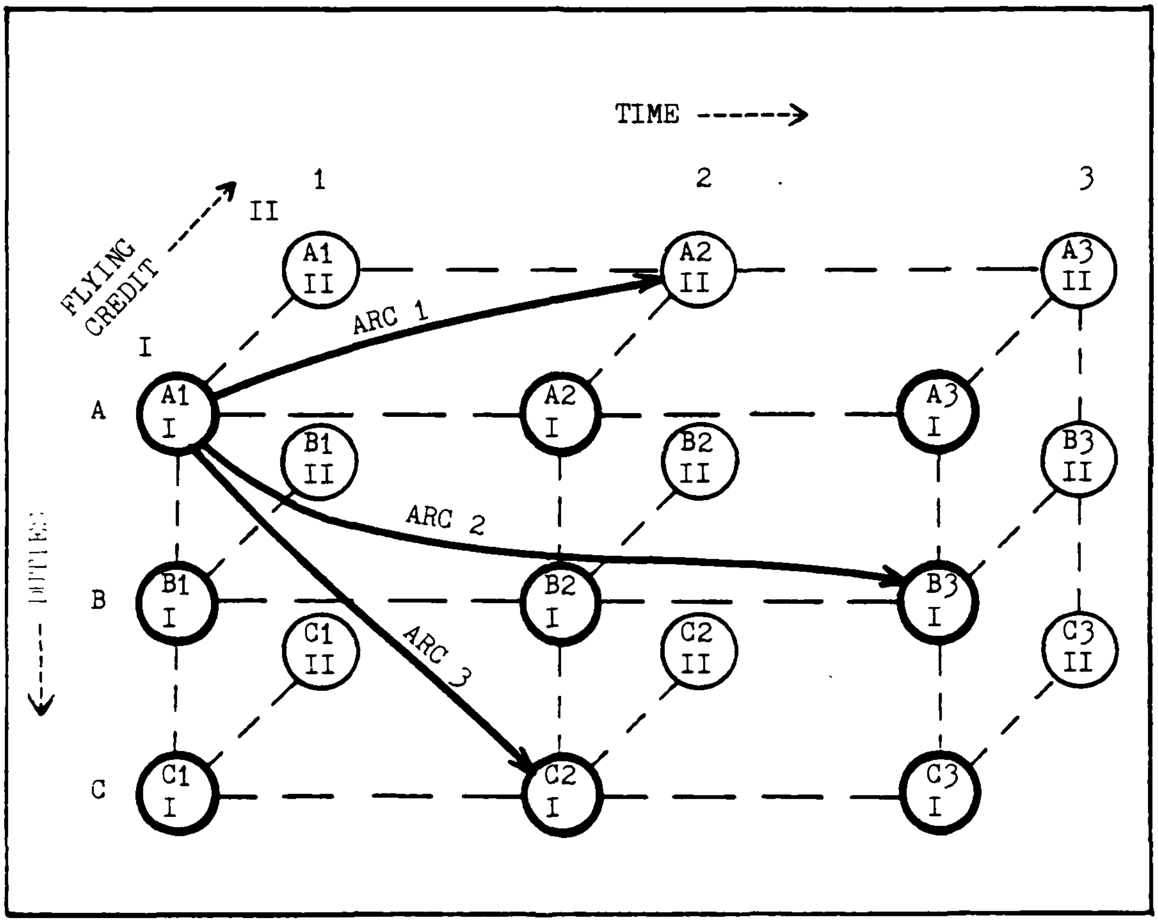


Figure 3. Three-Dimensional Assignment Network

represents, in this example, a potential change in gate credit from "I" to "II." (The dashed lines in the figure do not represent assignment paths--they are provided only to help visualize the three-dimensional nature of the figure.) The assignments represented by arc 1, arc 2, and arc 3 are the same as previously described for Figure 2. However, if we assume that duty type A is a flying duty and therefore results in an increase in gate credit equivalent to the assignment duration, individuals assigned to duty A via arc 1 acquire an increase in gate credit from "I" to "II". Note that duty types B and C represent nonflying duties in this example, since assignments along arcs 2 and 3 result in no increase in gate credit.

The fourth dimension--consisting of aviation service date year groups--cannot be illustrated graphically along with the other three dimensions, but it is not difficult to conceptualize. If an individual's ASD year group is taken to mean the number of years since the initial date of aviation duty, it can be seen that ASD year group for an individual progresses directly with time. In terms of the network structure, this means that any assignment path will terminate at a node representing an increase in ASD year group corresponding to the duty assignment duration.

The constraints on the assignment process can be represented by various network techniques such as nodal external flows (gains and losses to the system) and non-network side constraints (for example, total manning and experience level requirements for each duty type).

Computerized Network Model

This study resulted in development of a computerized network model of the Air Force rated officer assignment system. The model is a single commodity network flow model with side constraints. The modeling routine, named GATES, is written in FORTRAN 77. It essentially builds a network structure to describe the rated officer assignment system. A network optimization routine named NETSID (20) is employed to determine a set of assignments (arc flows) that minimizes a cost-weighted sum of individuals failing to meet their flying gates. A detailed user's guide for running the GATES routine is contained in Appendix A.

Modeling Routine. GATES provides a means for input of data and parameters, builds the network structure of the assignment system, specifies constraints on the network, and produces output data files for use by the optimization routine. After the data files are built, GATES calls the network optimization routine (NETSID). Upon completion of the optimization process, GATES calls an output routine which converts the NETSID optimal flow solution to information that is usable by AFMPC personnel managers.

The GATES modeling routine was written to deal with only one major weapon system group and one aeronautical rating. Separating the rated officer force by aeronautical rating (pilot or navigator) and major weapon system group results in a smaller network model for each of the groups. Since there is very little cross flow between weapon system groups and aviation ratings, little realism is lost by treating these groups separately. This study focused on strategic airlift pilots. However, GATES is designed to handle other major weapon system groups and aeronautical ratings by adjusting input parameters. GATES consists of approximately 2800 lines of FORTRAN computer code, including numerous internal comment lines. A complete listing of GATES is in Appendix B.

Network Solution Algorithm. The optimization routine employed in this study is a FORTRAN program called NETSID, provided by Dr. Jeff Kennington of the Operations Research Department at Southern Methodist University. NETSID was developed as a part of Dr. Keyvan Farhangian's doctoral dissertation (20:11).

NETSID is based on a specialization of the primal simplex linear programming algorithm which provides an efficient means of dealing with side constraints (19:166-167; 20:11). The algorithm employed in NETSID provides an increase in efficiency over a general linear programming routine when the number of nodes in the network is at least 10 times the number of side constraints (20:1-2).

The network problem with side constraints that is solved by NETSID can be stated mathematically as follows:

$$\text{minimize} \quad cx \tag{1}$$

$$\text{subject to} \quad Ax = r \tag{2}$$

$$Sx \begin{matrix} \leq \\ = \\ \geq \end{matrix} b \tag{3}$$

$$0 \leq x \leq u \tag{4}$$

where

A is the node-arc incidence matrix (each row represents a node, each column represents an arc, and the only nonzero entries in the matrix are 1 if the arc flows out of the node and -1 if the arc flows into the node),

S is a side constraint matrix (each row represents a particular side constraint, each column represents an arc, and the only nonzero entries in the matrix represent the side constraint coefficients--that is, the multiplicative factor indicating how much a particular arc flow contributes towards the right hand side of the side constraint),

b is the right hand side vector for the side constraints,

c is a vector of costs (where each nonzero entry represents the cost of a unit of flow on that arc),

r is the right hand side vector for the network (that is, the requirements demanded at each node),

u is a vector of upper bounds for the arc flow variables (that is, the maximum flow associated with individual arcs),

x is the solution vector of arc flows.

Actually, NETSID is capable of solving problems with more complex formulation, but the mathematical formulation shown here is adequate for treatment of the rated gate management problem. (19:166; 20:1)

As used in this study, NETSID reads data files created by GATES, determines an optimal set of arc flows based on minimizing the total value of the objective function (subject to conservation of flow requirements, arc capacities, and the side constraints) and produces an output data file consisting of the optimal solution arc flows.

Model Assumptions.

As with any representation of a real world system, there are several assumptions made in this study to permit development of the GATES model. The assumptions affect the structure, size, and detail of the network model, as well as the collection of input data and interpretation of the output data.

Perhaps the most significant assumption behind the GATES model is the assumption that the level of aggregation inherent in the model provides sufficient detail to provide the necessary insight into the rated gate management problem. The aggregation assumption permits treatment of the rated officer force as a large group of individuals whose attributes can be represented by the average attributes of the group as a whole.

The minimum time interval modeled in GATES is one year. Several assumptions inherent in GATES are based on this time increment. Some of the key assumptions are as follows:

> 1. Duty assignment tour lengths for the duty types modeled in GATES can be adequately represented as whole-number multiples of one year (and all individuals serve in a duty for "precisely" this amount of time).
>
> 2. All duty rotations, and therefore all assign-

ments made during a year, can be thought of as occurring at the mid-point of the year.

3. The overall manning requirements applicable to a particular duty type and year are applied at the end of the year--after all assignments for that year have been made.

4. Individuals who are projected to reach a particular ASD year group or a particular level of flying gate credit by the mid-point of the year can be considered to possess new ASD or gate credit at the time of rotation.

Other assumptions are required due to the nature of the network method employed. For example, attrition of officers from the Air Force occurs throughout the expected assignment duration. However, the network structure that NETSID is capable of solving can allow losses (and gains) to occur only at nodes of the network. Within the GATES model, an "assignment" of an officer to a new duty position is represented by flow on an arc of the network. Essentially, an individual is flowing on the arc throughout the average duty duration and does not reach a node until the end of that duty assignment. As a result, the GATES model assumes that all attrition losses occur at the end of average assignment durations.

Additional assumptions inherent in the network structure specified in GATES are described below.

Model Structure.

As discussed above, within each aeronautical rating and each major weapon system the flow of rated officers through various duty assignments can be thought of as a four-dimensional network. Each node in the network has associated values for time, duty type, ASD year group, and flying gate credit.

The arcs represent assignments to a new duty. The differences in time, ASD year group, and gate credit between the ending node and the beginning node of an arc represent the change in an individual's status while serving in the associated duty assignment. The duty type associated with the arc's beginning node represents the duty position occupied prior to the assignment. The duty type at the arc's ending node represents the new duty position which is occupied after the assignment occurs.

To insure adequate representation of real world duty assignment options and to insure compliance with real world manning considerations, constraints are applied to various arc flows. Some of these constraints are internal to the structure of the network model, such as permitting only certain nodes to be connected via arcs. Other constraints are modeled through the use of side constraints.

Model Objective Function. The objective of this model is to identify an assignment policy which results in maximum attainment of flying gates while also meeting constraints on manning and experience levels within various duty types. This objective is modeled by associating "cost" values with flows on certain arcs. These arcs are the ones that lead to nodes with associated ASD year group and gate credit values representing a failure to meet a particular flying gate.

For example, arcs with end-node ASD values equal to 12 years or greater and gate credit values less than six years have an associated cost because assignment (flow) along these arcs indicates failure to meet the first (six-year) gate. The network optimization routine determines the set of arc flows that results in minimum total cost (maximum gate attainment) while complying with the specified constraints.

The actual costs assigned to the various arcs could have considerable impact on the model solution. Air Force Regulation 36-20 provides some basis for assignment of costs (8:12). The regulation indicates a great deal of emphasis on attaining the first (six-year) and second (nine-year) flying gates. There seems to be slightly less emphasis on completing the third (eleven-year) gate. This suggests equal costing of arcs resulting in failure to meet the first and second gates. A slightly lower cost might be applied to arcs resulting in failure to meet the third gate.

Network Node Parameters. Each node represents a particular combination of time, duty type, ASD year group, and gate credit. The number of nodes in the network and the number of arcs connecting them greatly affects the amount of computer run time required to solve the network. It is fairly easy to exceed practical limitations on the size of a network that can be solved using an iterative solution routine such as NETSID. Large problems usually require many iterations, which can lead to numerical problems due to computer round-off errors. Specification of the four parameters identifying each node determines the overall size of the network for GATES. These four parameters are built into the GATES modeling routine.

Time Periods. The first dimension that is required to build the network model is the time dimension. Because a different set of network nodes is used to represent each time period that is modeled, the total number of time periods modeled has a large effect on the size of the network. There are actually two time related decisions that must be made:

1. What is the smallest time increment that should be modeled?

2. What should be the total time horizon of the model?

The answer to the first question depends on the level of detail deemed necessary. The answer to the second question depends on how far into the future one wishes to model.

In determining the smallest time increment to use, an important consideration is the smallest time period that can "satisfactorily" model duty durations. The smallest time increment used also affects two other dimensions of the model--ASD year group and flying gate credit--because ASD year group is incremented by all duty assignments and flying gate credit is incremented by assignment to flying duties.

Discussions with the AFMPC Analysis Division resulted in an initial selection of 0.5 years as the preferred time increment (34). Experimentation with the resulting network model indicated a need to increase the time increment to 1.0

years due to excessive model run times. Subsequent changes in the model structure have reduced the network size, potentially permitting a return to 0.5-year increments. However, GATES as presently formulated uses a 1.0-year increment.

The selection of a model time horizon for GATES also involved several considerations. Too short of a time horizon can result in insufficient representation of long-term effects. However, the changing nature of manning requirements and the force structure suggests that excessively long time horizons may result in long-term predictions based on invalid assumptions. Much planning data important to this model is predicted out for the next five years as basis for the Five Year Defense Program. The availability of this data--as published in the Rated Management Document--resulted in selection of a five year time horizon (9).

Because the estimated average duration of some duty assignments is approximately four years, a five year horizon may result in "optimal" solutions that fail to account for the effect of long duration assignments. When a duty assignment such as AFIT (1.5 years) and a follow-on Rated Staff/Supplement tour are considered, the total effective duration of assignments outside the cockpit can be five years or longer. This limitation of a five year horizon must be recognized when interpreting model outputs.

If a five-year time horizon is found to be insufficient, one solution may be to extend the time horizon for an additional two to three years and apply the planning data (manning requirements) from year five to all subsequent years.

Since much of the input data for the GATES model is based on fiscal year planning, fiscal year begin and end dates are a logical choice for the beginning and ending times for each modeled time period.

Duty Types. Duty assignment type is the second dimension used to identify nodes in the GATES network model. The actual variety of rated officer duty assignments is extensive. However, only a few factors need to be examined in determining the level of duty type aggregation that is appropriate to the rated gate model:

1. Contribution towards gate credit (flying versus nonflying duties);

2. Average duty duration (tour length);

3. Preceding and follow-on assignment possibilities.

These characteristics determine which nodes are connected together by assignment arcs in the network model. For instance, whether a duty assignment results in gate credit accumulation determines if the end-node gate credit value is the same as the begin-node value (nonflying duties) or is incremented by an amount equal to the duty duration (flying duties).

Consideration of these characteristics resulted in initial identification of five duty categories: operational flying, advanced student (flying), rated supplement and (nonflying) staff, Air Force Institute of Technology (AFIT) graduate student (including civilian institution programs), and resident professional military education (PME). However, it was difficult to separate advanced students from operational flying duties when the initial data was extracted from the AFMPC database. As a result, advanced students were included with operational flying duties. Since operational flying is the only likely assignment following duty as an advanced student, the only adjustment required by combining the two duty types was an increase in the average duty duration for operational flying.

The resulting four duty assignment categories, brief descriptions, and their estimated assignment durations to the nearest half-year increment are as follows:

> 1. Operational Flying; includes all flying duties within the major weapon system, including Advanced Student, and results in gate credit accumulation; 3.5 years.
>
> 2. Rated Staff/Rated Supplement; includes all nonflying duties except those included in "AFIT" and "PME"; 3.5 years.
>
> 3. AFIT; includes all full-time graduate degree programs; 1.5 years.
>
> 4. PME; includes Air Staff Training (ASTRA) and all resident PME courses requiring a permanent change of station (this includes Intermediate Service School and Senior Service School, but not Squadron Officer School); 1.0 years.

All estimated duty durations are whole number multiples of 0.5 years, but they are generally not whole number multiples of 1.0 years. Thus, selection of a 1.0-year minimum time increment causes some potential problems with realistic modeling of duty durations. One way to deal with this is to make model runs using duty durations which bracket the estimated average duration, such as making runs at 3 years and 4 years for operational flying assignments, and compare the results.

Aviation Service Date Year Group. The third dimension identifying GATES network nodes is the aviation service date (ASD) year group. For the purposes of this study, individuals who began aviation service within the same one-year period belong to the same ASD group. During model time period one, those who began flight duties within the year immediately preceding the first year modeled are in ASD year group 1; those who began flying two years before model year one are in ASD year group 2; and so on. An individual's ASD year group increases as time progresses, since ASD represents the number of years since the initial aviation service date (for the GATES model).

Grouping by aviation service date provides a ready means for monitoring compliance with flying gate requirements because these requirements are based on aviation service date. Since it is Air Force policy to keep rated officers in operational flying assignments continuously from initial date of aviation service until they have accumulated

6 years of flying gate credit, the first ASD group of interest to this study is ASD group 6. However, ASD groups below 6 are also modeled, since individuals in these low ASD groups will progress to higher ASD groups during the model time horizon.

Since the last flying gate must be met by 18 years of aviation service, ASD group 18 is the last group of interest in terms of gate management. Individuals belonging to ASD year groups greater than 18 are included with ASD group 18 in the GATES model.

Since a 1.0-year time increment is used in GATES, ASD groups are also incremented by multiples of 1.0. Including those who have not yet reached the one-year group, there are a total of 19 ASD year groups in the gates model (zero through 18).

Flying Gate Credit Accumulated. The final dimension that identifies each network node represents the amount of flying gate credit that has been accumulated. The tracking of flying gate credit is essential to this study. By comparing gate credit accumulated with ASD year group at each node, it can be determined which nodes represent failure to meet flying gates. Costs can be assigned to the arcs leading to these nodes. Since the network optimization routine will attempt to find a solution which minimizes the total cost of arc flows, the final solution will generally specify assignment to flying duties for those individuals who must fly in order to meet gate requirements.

Based upon the Air Force policy of flying rated officers continuously for the first 6 years of rated service, this study assumes that all rated officers accumulate gate credit equivalent to their ASD year group until six years of aviation service. Since young aviators with gate credit amounting to less than six years may transition above the six year group during the model time horizon, gate credit values down to zero are modeled by GATES.

Once an officer has accumulated 11 years of gate credit, the last flying gate requirement has been met. Many rated officers with over 11 years of accumulated gate credit continue to fill cockpit positions, but these individuals are included in the 11 year gate credit group in the GATES model.

Since the minimum time increment used in GATES is 1.0 years, gate credit can only be accumulated in multiples of one year in the model. Therefore, GATES includes 12 values of gate credit accumulation: zero through 11 years.

<u>Network Dimension</u>. The overall size of the network can be estimated by multiplying the number of values modeled for each of the four node parameters: (5 time periods) x (4 duty types) x (19 ASD groups) x (12 gate credit values) = 4,560 nodes. Actually, the number of nodes in the GATES model is significantly less than this because some of the nodes do not exist for certain combinations of ASD year group and gate credit values. GATES, as applied in this study, builds a network containing 975 nodes.

Note that a change in the minimum time increment used affects three dimensions of the network: time, ASD year group, and flying gate credit accumulated. There is little point in decreasing the time increment without also decreasing the increments for ASD year group and flying gate credit accumulation. If 0.5-year time increments were used in GATES and all else remained the same, there would be nearly eight times as many possible nodes: (9 time periods) x (4 duty types) x (37 ASD groups) x (23 gate credit values) = 30,636 nodes. Again, the number of nodes modeled could be reduced due to various constraints.

Network Arcs. The arcs of the network connect only certain combinations of nodes. Flow on an arc can be thought of as representing time spent in a duty assignment. Immediately before beginning a new duty assignment, an individual's current status is represented by the parameters of the node at the beginning of the "assignment" arc. The parameters describing the arc's ending node have two meanings: the ending node's duty type parameter indicates the duty occupied immediately upon arc entry; the remaining parameters of the arc's ending node represent the status of the individual upon completion of the duty assignment (the accrued time period, ASD year group, and gate credit). The rules used by GATES to determine which nodes are connected by arcs are discussed below in the section on structural constraints. As implemented in this study, GATES builds a network containing 2,374 arcs.

External Flows. Besides the flows that occur along the arcs of the network, the GATES model provides for "external" flows--flows that enter or leave the network. A network requires nodal conservation of flows. That is, flows into each node must equal flows out of each node. Generalized to the entire network, this means that external flows into the network must equal external flows out of the network. Flows entering the network at a node are called "positive external flows." Flows leaving the network at a node are called "negative external flows." (16:2)

In the rated gate management model, external flows are used to represent officer rotations into the network, to account for attrition, and to represent manning requirements extending beyond the modeled time horizon. The rotations into the network are modeled with positive external flows, whereas attrition and end-of-time-horizon requirements are modeled with negative external flows.

Rotations and Gains. Positive external flows are used to initialize the network. Basically, individuals enter the network at nodes representing the expected time of their next duty assignment rotation. This time of initial rotation is projected by examining the dates that individuals arrived at their current duty station and adding the average duration for that particular duty as modeled by GATES. The node at which individuals enter the network is further defined by the current duty type, the projected ASD year group (as of the time of projected rotation), and the projected gate credit level.

Rotation data is extracted from the AFMPC data base and is input to GATES from a data file. The data is aggregated by time periods (representing a range of date-arrived-station values), current duty types, ASD year groups, and gate credit amounts.

Positive external flows are also used to input expected gains to the major weapon system group. The rotation data applies only to individuals who are already identified with a particular major weapon system group and aeronautical rating (strategic airlift pilots were modeled in this study). Additionally, the Rated Mangement Document (9) contains data on expected gains to the major weapon system for the next five years. These gains include new graduates of undergraduate flying training programs. Gains also include individuals who served their first flying duty as flight instructors with Air Training Command and have not yet been identified with a major weapon system group (often referred to as FAIPs--First Assignment Instructor Pilots).

Attrition Losses. Negative external flows are used in GATES to model losses from the major weapon system group due to attrition (such as retirements and separations from the service). The AFMPC Analysis Division estimates the expected retention (continuation) rate for each major weapon system group, aeronautical rating, and ASD year group. GATES applies these continuation rates to the total number of individuals in each ASD year group at the start of the modeled time horizon to determine how many from each ASD group are expected to be lost each year. This yearly attrition amount for each ASD year group is further divided among duty types and gate credit values to determine the negative external flow that should be assigned to each node.

In the GATES network, individual flows arrive at nodes only at the completion of a tour of duty. By using negative external flows to model attrition, GATES relies on an implicit assumption that attrition can occur only upon completion of a tour of duty.

Actually, attrition occurs throughout duty assignments. Conceptually, this is only a small problem since the total attrition demanded by the nodes representing a particular year and ASD group corresponds with the expected losses to that ASD year group. However, the number of individuals from each duty/ASD/gate credit combination that are available for rotation varies each year. As a result, for some situations there may be insufficient flows into a particular node to supply the negative external flow (attrition) demanded at that node. For situations when this occurs, GATES provides a simple means for varying the modeled distribution of ASD groups among duty types and gate credit values. By specifying positive and negative values of an input parameter (named "CHANGE"), the amount of attrition demanded at any particular node can be adjusted. Though this is a somewhat circuitous (and imprecise) means of obtaining a feasible solution, the distortion is probably not significant if care is taken.

End Requirements. The final node in the network is assigned a large negative external flow to provide an overall balance to the network. The external flow assigned to this end-of-network "sink" node is calculated by summing all rotations and gains into the system and subtracting all losses (attrition) from the system.

This node represents manning requirements occurring beyond the time horizon of the model. This sink node is fed by four nodes--one for each of the duty types--that collect all assignment flows that would terminate beyond the time horizon of the model. The assignment of a large negative external flow to this node helps enforce nodal conservation of flow.

Model Constraints. Model constraints provide a means for controlling the network solution so that results more closely conform to real world criteria. Some constraints are imposed upon the network structure within the GATES modeling routine through specification of nodes and arcs. Other constraints are modeled through the use of side constraints. The side constraints are identified in GATES and are used by NETSID during the optimization procedure. Side constraints are constraints that apply to flows across multiple arcs.

Structural Constraints. The structural constraints in GATES determine which combinations of duty type, ASD year group, and gate credit accumulation "exist" and therefore identify nodes in the network. Structural constraints also determine which nodes are connected via arcs. Unless two nodes are connected by an arc, there is no means (within the model) to transition between the states represented by the two nodes.

The following is a list of constraints imposed on the network structure:

1. The maximum ASD year group attainable is 18. Though inidividuals actually enter higher year groups, they remain in the 18 year group in GATES because gate requirements are tracked only until the 18-year point.

2. The maximum gate credit accumulation possible is equal to the ASD year group or 11 years, whichever is less. Though many individuals actually fly beyond 11 years of gate credit, there is no need to do so from the standpoint of gate requirements. Gate credit greater than 11 years is not differentiated from gate credit equal to 11 years in the model.

3. Flying duties result in accumulation of additional gate credit equal to the duration of the assignment, whereas nonflying duties result in no increase in gate credit. Thus, only certain combinations of nodes are connected by arcs representing the various types of duties. Arcs representing flying duty assignments lead to nodes indicating an increase of gate credit, while arcs representing nonflying duty assignments do not result in an increase of gate credit.

4. Individuals with six years or less of aviation service occupy only flying duties. As a result, their gate credit accumulation equals their ASD year group. Though there are a few real world exceptions to this, Air Force assignment policies minimize the number of exceptions (8:17).

5. AFIT assignments lead only to staff/supplement assignments. This conforms with the stated requirement to serve a minimum of three years in a nonflying duty following AFIT and policies against

assignment to PME until a minimum of three years following an AFIT assignment (8:45; 11:21).

6. AFIT assignments are available only to ASD groups less than 14 years. Very few rated officers in year groups above this attend full-time AFIT progams.

7. Individuals occupying a PME assignment cannot flow to another PME assignment. PME is a limited opportunity and back-to-back PME assignments are contrary to reality.

8. PME (including ASTRA) is available only to ASD year groups 6 through 8 (ASTRA), 12 through 14 (Intermediate Service School), and 17 and above (Senior Service School). This corresponds with the approximate eligibility periods for assignment to these special duties.

9. Attrition occurs only at Flying and Staff/Supplement nodes. This conforms with the real world service commitments that are incurred as a result of an AFIT or PME assignment.

In addition to these structural constraints that are built into the GATES modeling routine, side constraints are used to control the flows on certain arcs.

Side Constraints. GATES specifies side constraints for insuring manning and experience requirements are met. These side constraints result in a solution from NETSID that represents assignment of sufficient personnel to each duty type. GATES also provides a means of specifying the mix of individuals--by ASD year groups--that are assigned to a particular duty type.

The manning requirements for each of the four duty types modeled can be obtained from the Rated Management Document (9). GATES provides a means for specifying these requirements for each fiscal year. GATES also permits specification of undermanning and overmanning tolerances. These tolerances identify the percentage by which the NETSID solution is allowed to undershoot or overshoot the stated requirement for each duty type and fiscal year.

Essentially, GATES identifies all arcs which represent individuals occupying a particular duty type during a particular year and assigns the manning requirements for that year and duty as the associated side constraint. The NETSID routine insures that the total of all flows across the indicated arcs complies with the associated side constraint(s). Each manning requirement translates to two side constraints: a "greater-than" constraint which requires the sum of the associated flows to be greater than or equal to the undermanned requirement (the requirement reduced by the undermanning tolerance amount); and a "less-than" constraint which requires the sum of the flows to be less than or equal to the overmanned requirement (the requirement increased by the overmanning tolerance amount).

Experience level requirements can also be specified in GATES. These side constraints may be necessary to insure that particular duty types are manned by the right mix of individuals (that is, the NETSID optimal arc flows include sufficient flows along the proper arcs). For example, it is important that flying duties be filled by a sufficiently large percentage of "experienced" flyers. If no side constraint is specified, the NETSID solution may indicate assignment of insufficient experienced individuals to flying duties. By specifying side constraints, NETSID can be forced to produce a solution more representative of real world considerations. Otherwise, the optimization routine will always result in merely the "least-cost" solution.

The structure provided in GATES for input of experience level constraints is flexible enough to allow identification of as many constraints as is deemed necessary to force a realistic solution. The inputs required to specify each experience level side constraint are as follows:

1. The duty type;

2. The ASD year group that serves as the "cutoff" (the overall group being constrained is all ASD groups greater than or equal to the cutoff group, or all ASD groups less than or equal to the cutoff group, depending on the type of constraint);

3. The percentage of total manning within the duty type that must consist of the designated ASD year groups;

4. The type of constraint--"less-than" or "greater-than"--that applies (this indicates whether the percentage of positions filled by the specified ASD groups is a maximum or minimum requirement and also indicates whether the applicable ASD groups are greater-than or less-than the cutoff ASD).

Because different rules are used for applying less-than and greater-than constraints, some degree of flexibility is provided.

As an example of a "greater-than" experience level constraint, it could be specified that a minimum of 60 percent of all flying duty positions be occupied by individuals belonging to ASD year group six or higher. In determining which arcs are constrained by this requirement, GATES would examine all arcs terminating at the indicated duty type with an end-of-tour ASD group greater-than-or-equal-to six. The total of all flows on these arcs would have to be greater-than-or-equal-to 60 percent of the total manning requirement for flying duties. This constraint applies individually to each time period in the model.

As an example of a "less-than" constraint, it could be specified that a maximum of 40 percent of all flying positions be occupied by individuals belonging to ASD year group five or lower. In implementing this constraint, GATES would examine all arcs terminating at flying duties with beginning ASD groups less-than-or-equal-to five. The total of all flows on these arcs would have to be less-than-or-equal-to 40 percent of the total manning requirement for flying duties.

Because GATES uses the duty tour beginning or ending ASD year group in determining whether an experience level constraint applies to a particular arc, the change in ASD year group that occurs during a duty tour duration must be considered when specifying experience level requirements.

Model Inputs

The inputs to the model were obtained from the AFMPC personnel data base and the Rated Management Document, a two-volume publication updated annually that contains rated personnel planning data from the Air Force's Five Year Defense Program (9; 10).

Input Data File. The data that provides the bulk of the positive external flows into the network must be compiled from the AFMPC data base prior to model run time. The required format for the data file is specified in the GATES User's Guide (Appendix A).

The data is collected for rated officers assigned to a selected major weapon system group and aeronautical rating, for example strategic airlift pilots. Within this group, individuals are categorized by current duty type, ASD year group, and years of gate credit accumulated. For each of these duty/ASD/gate combinations, the data file lists by time period the total number of individuals that arrived at their current duty assignment. Using the chosen duty durations and the arrived-station date, the time of next expected rotation can be determined. By incrementing ASD and gate credit values accordingly, the node at which each group of individuals should enter the model can be determined.

Input Parameters. GATES is designed to provide a large degree of flexibility in specifying modeling parameters. The values of these parameters can be adjusted, allowing application of the model to a broad range of major weapon system groups. As presently implemented, GATES does not provide an external means of adjusting parameters. Instead, all changes must be made internally to the FORTRAN code and the file must then be compiled and linked to NETSID before running the program. However, all of these parameters are contained in a single section (Section II) of the GATES routine.

The following parameters can be easily adjusted within the FORTRAN code by changing the right hand side of the assignment statements in Section II of GATES:

1. The average duty duration, in years, for each of the four modeled duty types;

2. The costs assigned to arcs associated with non-attainment of gates (different values can be assigned for each gate);

3. The number of gains to the major weapon system for each year, broken out by gains from undergraduate flying training and gains of First Assignment Instructor Pilots (FAIPs);

4. The manning requirements for each of the duty types and each of the years;

5. The breakout of PME/ASTRA duties by the number of positions for each of the three categories--ASTRA, Intermediate Service School, and Senior Service School;

6. The amount of overall overmanning or undermanning (as a percentage of baseline manning requirements) allowable for each of the duty types and each year;

7. The continuation (retention) rates for ASD year groups 6 through 18 for each of the modeled years;

8. The distribution of personnel in ASD year groups 6 through 18 (by percentage) among the modeled duty types and gate credit values (this is used in calculating attrition external flows).

Besides adjustments to costs associated with not attaining particular flying gates, additional flying time goals--with associated costs for noncompliance--can be specified.

Also, a relatively simple means for specifying manning experience levels for any of the duty types is provided. It is assumed that such experience requirements apply across each time period individually. The overall experience level within a duty type must be expressed in terms of the minimum or maximum percentage of positions that are to be occupied by individuals from specified ASD year groups.

Model Outputs

The output files produced by the GATES model serve several purposes. Some of the files serve as temporary storage locations for data used in building the network and solving it; one file contains a presentation of the optimal network solution in terms of duty assignments; and other files contain auxiliary information useful in debugging model modifications and verifying the model logic structure.

Network Data Files. GATES produces four data files that serve to describe the entire structure of the gate management network, including external flows and side constraints. These files (named "FOR001.DAT" through "FOR004.DAT") are subsequently read by the NETSID routine during the optimization phase. Example lines from these data files are in Appendix C.

NETSID produces two additional files. "FOR007.DAT" contains optimization information such as the final objective function value and the number of iterations required, as well as a complete list of the solution arc flows. "FOR008.DAT" contains only the optimal solution arc flows;

it is subsequently read by GATES to produce a final solution output file. A sample of "FOR007.DAT" is in Appendix D.

Information Output File. The optimal assignment schedule, including identification of any assignments resulting in gate non-attainment, is contained in a file named "ROTEPLAN.OUT." It is simply a formatted output of the data contained in "FOR008.DAT." A sample of this file is in Appendix F.

Auxiliary Output Files. Two additional files are produced by GATES for use in debugging, validation, and verification. "SANITY.OUT" contains various information derived from the input data and from calculations performed by GATES (see Appendix E). "NODEARC.OUT" is simply a list of all network arcs with their beginning and ending nodes (see Appendix G). The four parameters identifying each of the nodes are also listed in "NODEARC.OUT."

Screen Output. When run interactively, GATES and NETSID output some information to the terminal screen. Screen output includes information useful in monitoring the performance of the routine, as well as general information about model inputs and the size of the network built by GATES. An example of the screen output is in Appendix H.

Model Run Times

The run times of the GATES model depend on the particular computer system being used. During the initial stages of model development, GATES was run on a Digital Equipment

Corporation VAX 11/785 mainframe computer. During periods of low computer usage, total run times for GATES were approximately 20 minutes and about 10 minutes of CPU time were required. During the final stages of model development, a VAX 8650 computer became available. The performance improvement was significant: total run time was reduced to about 4 minutes with about 2 minutes of CPU time required.

The actual run time varies with the number of iterations required by the optimization procedure, which depends on the particular parameters used. The number of iterations required varied between 1800 and 2500, depending on the parameters specified.

Model Verification

Model verification can be thought of as the determination of whether the model actually performs in the manner that it was intended to perform. Note that this is different from determining whether the model adequately represents the real world system; that determination is made during the validation phase.

In reality there is much overlap between verification and validation. Many analysts do not even distinguish between the two phases and treat them together. Forrester and Senge state that the "ultimate objective of validation...is...confidence in a model's soundness and usefulness as a policy tool" (13:211). Verification involves determination of the model's soundness. The model's usefulness (including the degree to which it represents reality) is discussed under the section on validation.

Since verification focuses on the model's internal soundness, the logic and structure that go into developing the model provide a reasonable basis for model verification. Forrester and Senge identify several methods for verifying model structure. Among these methods are the "structure-verification" test and the "boundary-adequacy" test (13:212, 214-215).

Structure-Verification Test. Forrester and Senge state that verification of model structure involves "comparing structure of a model directly with structure of the real system that the model represents" (13:212). The general idea here is that a model that is structured like the real world system will behave like the real world system. This is the attitude taken during development of GATES.

The degree of aggregation required to reduce the rated officer gate management issue to reasonable proportions necessitated some major simplifications during development of the GATES model. For example, the assignment process is examined only once each fiscal year (the impact of a one-year time interval in GATES). Although this assumption may hide some of the system fluctuations that rated officer managers need to know about, the level of detail still is sufficient to provide a significant improvement over the existing arithmetic model.

An area in which GATES seems to fail the structure-verification test is in its treatment of attrition. Given that attrition actually happens throughout the period of an assignment, the GATES constraint that attrition only occurs at rotation time seems to violate the real world situation. However, this is not a real problem. Given that attrition occurs throughout a duty assignment in the real world, assignment officers are sometimes required to "fill" vacancies that occur only partially into a full tour of duty in that position. By requiring attrition to occur only at the end of a full assignment duration, GATES does not allow for these intermediary assignments to occur. Over the broad picture, though, the number of required assignments should not change due to this modeling assumption. Furthermore, though the GATES assumption allows only rotating individuals to be attrited, the method of calculating attrition at each node helps insure a fairly realistic distribution of attrition across ASD year groups, duty types, and flying time.

Boundary Adequacy Test. The purpose of Forrester and Senge's boundary-adequacy test is to consider the "structural relationships necessary to satisfy a model's purpose. The boundary-adequacy test asks whether or not model aggregation is appropriate and if a model includes relevant structure" (13:214).

One factor that may readily test the adequacy of the modeled boundaries of GATES is the realistic average duty duration. Discussions with the AFMPC Analysis Division indicated a difficulty with determining average duty durations for flying duties and rated staff/supplement duties (34). Examination of the rotation data obtained from the AFMPC database indicates that the average duty duration for flying and staff/supplement duties is probably in excess of three years and may approach four years. Considering that the modeled time horizon is only four years, this aspect of the model boundary may bear further scrutiny. If the model's usage is limited to short term indications or very general long term effects, this point appears to be insignificant.

Model Validation

The validation phase of this study involved assessing the usefulness of the GATES model. As stated by Forrester and Senge, "It is pointless to try to establish that a particular model is useful without specifying for what purpose it is to be used" (13:211). The stated purpose of this study was to develop a model to help improve the effectiveness of rated officer force management. The final validation of GATES will occur if and when it makes a contribution towards increased effectiveness in gate management.

Some of the key tests that can be used in model validation are categorized as behavioral tests (13:217-223). Two of these tests identified by Forrester and Senge are the "behavior reproduction" test and the "behavior sensitivity" test (13:217-219,222-223).

Behavior Reproduction. Behavior reproduction tests include several methods of determining the extent to which behavior in the model matches real world behavior under similar conditions (13:217). This test appears highly applicable to the GATES model with reference to the experience level side constraints.

During initial phases of model development, GATES contained no provision for specifying minimum experience levels (based on ASD year groups) for specified duties. The initial solution outputs showed large numbers of individuals in the older ASD groups being assigned to nonflying duties once they had met their gates. This seemed to be somewhat contrary to the real world situation, but seemed consistent with the lack of any experience requirements in the flying duties (in the GATES model). Consequently, the optimal solution to these initial versions of GATES seemed to provide overly optimistic results concerning achievement of flying gates.

With the addition of an experience level constraint suggested by the AFMPC Analysis Division (35), the optimal solution resulted in more individuals not attaining their flying gates. Subsequent structural changes to GATES reduced the impact of the specified experience level requirement. However, adding a requirement for a certain level of experience in flying duties reduces assignment flexibility and may result in fewer people attaining all of their flying gates, as would be expected in the real system.

Another case of expected behavior reproduction occurred when the undermanning tolerance for flying duties was reduced. This resulted in a more restrictive side constraint on assignments to flying duties. As might be expected, the result was assignment of more individuals to flying duties.

Behavior Sensitivity. This test relates to examining model behavior in light of parameter changes (13:222). Since several parameters are easy to adjust in GATES, it lends itself well to sensitivity analysis. The major shortcoming in this regard appears to be the oversensitivity of negative external flows (attrition) to other changes. Due to the method of modeling attrition in GATES, attainment of a feasible solution for a particular set of inputs often requires adjustment of the attrition distribution. The same number of people are attrited overall, but the amount of attrition occuring at each particular node is changed. Further analysis of the sensitivity of GATES to parameter changes is contained in the next chapter.

Summary

This chapter has presented the methodology employed in an attempt to deal with the rated gate management problem. The aggregation required by the network model limits the amount of detail provided by the model. However, the GATES model incorporates some of the dynamics of rated force duty rotations and provides potential insight into the affect of these rotations on compliance with gate requirements. The next chapter provides a discussion of the results of several runs of the GATES model.

IV. Results

This chapter presents the results obtained from the GATES network model. The chapter begins with a discussion of considerations relevant to interpreting network solutions. The discussion continues with an analysis of the solutions provided by GATES, including an examination of the sensitivity of GATES to various input parameters. The chapter concludes with a treatment of the practical implications of the results.

Solution Considerations

An analysis of results obtained from an optimization procedure such as the NETSID routine must include consideration of some of the peculiarities of optimization procedures based on linear programming methods. These peculiarities include the potential for multiple optimal solutions and infeasible solutions. Additionally, the sensitivity of the solution to changes in the input variables is of practical importance.

Multiple Optimality. Optimization procedures such as linear programming and its specializations, including network programming, are designed to produce the "best" solution possible based on maximizing or minimizing the stated objective function of the problem. In some situations, more than one set of solution values will produce the same optimal objective function value. In these cases, an optimization procedure such as network programming will generally identify only one solution. There are methods available for identifying all alternative optimal solutions for some linear programming formulations, but these are not directly applicable to the GATES/NETSID formulation (30:99-114).

For a cost minimization problem where a lower bound of zero is placed on the prospective minimal cost, such as in the GATES model, a solution which produces an objective function value of zero is likely to be only one of several possible minimum cost solutions. A similar situation could occur even when the optimal objective function value is not zero.

Interpretation of any results from GATES must consider the possibility of multiple optimal solutions. A zero-cost objective function seems likely if the number of flying duty positions exceeds those required to allow all rated officers to meet their flying gates--as may be the case in some major weapon system groups. For the inputs used in this study the resulting objective function was consistently greater than zero, but the possibility of multiple optimal solutions still exists. In some cases the objective function costs could result entirely from individuals who enter into the network model already in positions from which they cannot meet flying gates. The optimal solution presented could be just one of many which would prevent any additional costs beyond those that are already unavoidable. Examination of model outputs can help identify situations such as this.

The addition of increasingly more side constraints on the problem will tend to reduce the number of alternative optimal solutions, since constraints reduce the region of feasible solutions. If the analyst suspects the existence of alternative optimal solutions, addition of side constraints that increase the realism of the model outputs may be appropriate.

Infeasibility. Solutions to linear programming problems which include artificial variables in the basis solution set are "infeasible" solutions. In other words, such solutions do not exist within the specified bounds of the problem. An artificial variable is a variable that does not really exist but must be created in some cases by the optimization procedure to permit the procedure to produce a solution to the problem. Examination of the NETSID computer code and solution output indicates that NETSID assigns a cost of 20 or 40 to artificial variables. (Costs assigned to arcs leading to gate non-attainment should therefore be less than 20.)

A set of input parameters that results in an infeasible solution can still provide some valuable information. When GATES and NETSID produce an infeasible solution, the output files must be examined to determine the degree of infeasibility. If the solution arc flows indicate that relatively few units of artificial flow are created at some nodes in order to produce a solution, the artificial flows can be considered negligible in light of the aggregate nature of the model.

Infeasible solutions can result for several reasons, including the following situations:

1. Overall demand exceeds supply (attrition plus end-of-network sink value exceeds gains plus rotations into the network);

2. Overall supply exceeds demand (opposite of the above);

3. The demand at a particular node, such as the attrition demand, exceeds the available flow into that node;

4. The supply at a particular node exceeds the flows that can leave that node;

5. Side constraint limitations are exceeded.

GATES performs calculations and internal checks to help prevent situations leading to infeasibility. The end-of-network sink value is determined within the program by subtracting attrition from rotations and gains into the network. This should prevent occurrence of excess overall supply or demand. If other infeasible conditions are identified, program execution is stopped automatically prior to initiation of the optimization process. However, not all potentially infeasible situations are included in these checks.

The inputs examined by GATES during the feasibility checks include the rotations into the network (based on the input data file, "ROTE.DAT"), the gains to the major weapon system (from undergraduate flying training and First

Assignment Instructor Pilots), the number of manning requirements for each modeled time period, and the amount of attrition for each year. If requirements exceed availability, program execution is stopped and an error message is displayed on the terminal screen. These checks account for a large number of potentially infeasible situations. However, the method of attrition modeling employed in GATES is not completely incorporated into the checks.

Because attrition is calculated separately for each node, the solution flows (and, therefore, the available flow at each node) are not known prior to model run time. Thus, the feasibility checks can not prevent all possible infeasible situations. In these cases, GATES proceeds to call NETSID and the solution contains artificial flow variables (indicating infeasibility). When an infeasible solution results, information on the artificial variables (flows) is output to file "ROTEPLAN.OUT". If deemed necessary, adjustments can be made to the attrition values in an attempt to achieve a feasible solution. The means provided in GATES for attrition adjustment is a parameter (named "CHANGE") which can be used to increase or decrease the attrition demanded at any particular node. Sometimes only slight adjustments are required to achieve feasibility. Other situations require numerous model runs with attrition adjustments between each run. By using some care in redistributing attrition, excessive distortion can be avoided.

Sensitivity. A solution's sensitivity to model input parameters is another important consideration in interpreting the output of an optimization model. Especially when the real world values of some parameters are unknown, it is worthwhile to make several model runs while varying the input parameters. When some parameters are varied, they produce large changes in the model outputs. The sensitivity of the model outputs to changes in a parameter's value indicates the degree of importance that should be placed on further investigation of the correct value of the parameter to input to the model.

Solution Analysis

In this study, the GATES model was run with several combinations of input parameters in an attempt to establish some of the key factors affecting the model results. During these model runs, some factors were found to greatly influence the solution. In fact, altering some input parameters immediately resulted in an infeasible solution. The input parameters considered in the sensitivity analysis of the GATES model were the following:

1. Duty assignment tour lengths;
2. Costs of not attaining flying gates;
3. Experience level requirements;
4. Manning requirements;

5. Attrition;

6. The initial input data.

Listings of the key inputs and outputs from the sensitivity runs are in Appendix I.

Duty Durations. The average tour of duty must be input to GATES for the four modeled duty types (flying, rated staff/supplement, AFIT, and PME/ASTRA). The minimum time increment modeled in GATES is one year. Therefore, the estimated tour lengths must be rounded off to whole-number multiples of one year.

Because of the large manning requirements for flying and staff/supplement duties, errors in modeling these tour lengths can have potentially great effect on the model solution. Initially, the GATES model was developed using tour lengths of three years for both of these duty types. As discussed in Chapter 3, the estimated average tour length is between three and four years for both of these duties. For this reason, some model runs were performed with tour lengths set at four years for both duties.

Direct comparison of model outputs resulting from three-year tour lengths (for flying and staff/supplement duties) with outputs resulting from four-year tour lengths is difficult. This is because it was necessary to adjust attrition in order to achieve feasibility, and the adjustments required by the two situations were different. In fact, feasibility was not achieved for the four-year case. (The output was accepted as "close enough" when the artificial flow was reduced from 193 units to 10 units.)

The inputs for runs A1 and B1 (see Appendix I) were identical except for flying and staff/supplement tour lengths (three years for run A1 and four years for run B1). Since tour lengths affect the years in which individuals are rotated into the network model, the distribution of these rotations across the model time horizon varies between the two runs. Table 1 contains the appropriate information.

The flow of individuals into the network appears to be much smoother for the four-year case. For the three-year case, the number of individuals rotating into the model in year one is about twice the value for years two and three. This uneven input distribution is reflected in the number of assignments that are made in each of the model years. For the three-year case, many more assignments are made in years one and four than in the other years. Even in the four-year case there are about 20 percent more rotations input at year one and this results in more assignments made in years one and five than in the other three years. A relatively even distribution of assignment decisions over time is representative of the real world assignment process. Based on these results, it appears that four years (versus three years) may be a more realistic approximation of the average tour length for flying and staff duties.

Longer modeled tour lengths pose problems for the GATES model because of the means used for modeling attrition.

Model run:	A1	B1
Tour lengths (years)		
Flying duties:	3	4
Staff duties:	3	4
Rotations into Network		
Year 1:	1795	999
Year 2:	831	796
Year 3:	783	831
Year 4:	0	783
Year 5:	0	0
Total Assignments Made		
Year 1:	1797	1031
Year 2:	1062	958
Year 3:	930	863
Year 4:	1570	894
Year 5:	1146	980

Table 1. Tour Length vs. Rotation/Assignment Distribution

Because of the model's structure, individuals can leave the network only upon arrival at a node representing completion of a tour-of-duty. Long tour lengths therefore "protect" individuals from attrition throughout the duty tour length. This may not be a problem when individuals are evenly distributed among ASD year groups, gate credit time, and duty types. However, these distributions are not generally uniform. Especially in the early years of an ASD year group's aviation service, individuals in the year group tend to rotate together as a large block. This causes problems when attrition demands are evenly distributed across model years. As a result, long tour lengths require more adjustments to attrition in order to achieve a feasible solution. The data for model runs A2 and B2 (Appendix I) shows that fewer adjustments were required for the three-year case than for the four-year case.

Since gate completion is the goal of the GATES model, comparisons of model runs should examine the number of assignments (flows) leading to missed flying gates. Table 2 shows the relevant data. The rotation flows that are input

Model run:	A1	A2	B1	B2
Feasible Solution?	NO	YES	NO	NO
Tour lengths (years)				
Flying duties:	3	3	4	4
Staff duties:	3	3	4	4
Inputs That Already Missed Gates				
2nd Gate:	153	153	105	105
3rd Gate:	107	107	169	169
Assignments Resulting In Missed Gates				
2nd Gate:	0	143	0	49
3rd Gate:	107	363	189	392

Table 2. Tour Length vs. Gates Missed

to the network are entered at nodes based on projected ASD and gate status at the time of network entry. These projections are based on the modeled duty duration. This is why the number of missed gates input to the model is different for the two tour length situations.

For all four model runs shown in Table 2, the total number of optimal solution assignments leading to missed gates is greater than the number of missed gates initially input to the network. However, nearly all of the missed-gate assignments are for ASD year group 18 (or higher). The modeled duty duration for ASD year group 18+ is always one year (to allow sufficient flow to all ASD 18 nodes to meet the high retirement-induced attrition demands of these nodes). If ASD 18 assignments were for a full three or four years, depending on the case, many of these assignments would result in gate completion. Instead, GATES continues to reassign these individuals annually. As can be seen from the data listed in Appendix I, the number of missed gates tapers off in the later years of the model time horizon. In light of this, the GATES model solution generally results in fewer missed flying gates than are present in the input data.

It appears that GATES provides an assignment policy that leads to gate compliance that is more optimistic than the real world situation. The GATES solution reflects the best level of gate attainment that is possible, subject to the input parameters, the modeled constraints, and the relative costing of the various gate goals. If the parameters and constraints reflect the real world situation, then the GATES solution provides insight into potential problems with gate compliance and indicates an assignment policy that will minimize the degree of gate non-attainment.

Gate Failure Costs. An analysis of the effects of costs assigned to gate non-attainment is not hindered by the infeasibility problems caused by attrition. Once a feasible solution is achieved for a particular set of tour lengths (and other input parameters), altering the arc costs does not result in subsequent infeasibility. Model runs A2 through A9 demonstrate the effects of different gate costs. Table 3 contains the key inputs and outputs for these runs. More complete data is in Appendix I.

Varying gate costs have an indeterminate effect on the number of iterations required by NETSID to achieve the optimal solution. No particular relationship between gate costs and iterations is obvious.

Varying gate costs do have an obvious effect on the optimal objective function value: changing the costs associated with arc flows contained in the optimal solution set results in predictable changes in the total objective function value. For the inputs provided to GATES in model runs A2 through A9, the optimal solution consisted of some assignments resulting in failure to complete the second or third gates. Thus, changing the costs associated with the second and third gates resulted in changes in the objective function value.

Varying the cost of failing to meet the first gate, relative to the other gates, had no effect on the solution assignments resulting in missed gates. Also, reducing to zero the cost of assigning ASD year groups below six

Model run:	A2	A3	A4	A5	A6	A7	A8	A9
Costs								
Gate 0:	4	8	3	3	3	3	0	3
Gate 1:	3	6	3	3	3	3	3	3
Gate 2:	2	4	2	3	3	2	2	4
Gate 3:	1	2	1	1	3	2	1	1
Iterations:	1917	2046	1917	1899	2032	1751	1917	1845
Objective Cost:	649	1298	649	792	1518	1012	649	935
Assignments Resulting In Missed Gates								
Miss Gate 2								
To Fly:	143	143	143	143	143	143	143	143
To Staff:	0	0	0	0	1	1	0	0
Miss Gate 3								
To Fly:	363	363	363	363	362	362	363	363
To Staff:	0	0	0	0	0	0	0	0

Table 3. Gate Costs vs. GATES Model Results

("gate 0" in Table 3) to nonflying duties resulted in no change in the assignment solution. Both of these results could be attributed to the model structure. The network is structured to allow ASD year groups six and below to be assigned only to flying duties. Thus, completion of the first gate is "guaranteed" by the model. The "gate 0" cost was introduced in early development of GATES to prevent end-of-network flows to nonflying duties for these young ASD year groups. Subsequent structural changes to the model have eliminated this need.

The only important change that occurred with varying gate costs resulted when costs for gates two and three were set equal (model runs A6 and A7). Even then, the change was relatively minor: one individual was assigned to a nonflying duty that resulted in failure to meet the second gate. When the cost for gate two failure was higher than the cost for gate three failure (e.g., run A2), this same individual was assigned tc a flying duty resulting in meeting the second gate (but failing to meet the third gate).

The insensitivity of the GATES solution to changes in gate costs indicates that the assignments resulting in missed gates were due to the status of individuals at initial input to the network. In other words, several individuals entered the network at nodes indicating that they had already missed flying gates. Subsequently, the model generally assigned these individuals to flying duties until they accrued sufficient flying time to represent gate attainment.

Experience Requirements. The GATES model provides a means for specifying officer experience mixes, based on ASD year group, for any of the duty types. The intent of this provision is to allow for a means of preventing solutions that represent unrealistic assignment policies. Based upon a recommendation provided by the AFMPC Analysis Division, an experience level requirement was specified for flying duties (35). The requirement applied stipulates that a minimum of 50 percent of the flying duty positions must be filled by personnel possessing an ASD value of six or higher (as of the end of the tour of duty).

Model runs A2 and A10 contained the same inputs except that run A10 did not include the experience requirement. Comparison of the outputs from these two runs indicates very little change in the distribution of ASD year groups among duty assignments. This lack of effect due to the experience level requirement may be largely due to structural constraints that require all individuals to perform flying duties until the six-year point (thus preventing assignment of these individuals to nonflying duties). Additionally, this experience level constraint seems to impose no real hardship on the model. Regardless, the experience level constraints should be a useful feature for insuring the proper makeup of various duty types.

Manning Requirements. The manning requirements input to GATES came directly from the Rated Management Document (9). No sensitivity analysis was performed on the baseline manning requirements. However, the sensitivity of the model to overmanning and undermanning tolerance specifications was examined.

GATES performs internal checks to determine whether sufficient individuals are available to meet overall manning requirements specified in the parameters section of the program. If insufficient (or excess) personnel are available, execution is aborted and the NETSID optimization routine is not called since the result would be an infeasible solution.

Model runs A11 through A14 involved testing the impacts of manning tolerances. Model run A2 specified 10 percent undermanning and overmanning tolerance for flying and staff/supplement duties. For run A11, a five percent undermanning tolerance for flying duties was specified for all five model years. However, this resulted in failure of the "sanity" checks due to insufficient personnel in year five. For run A13, a similar result occurred when a five percent undermanning tolerance was specified for staff/supplement duties. Runs A12 and A14, respectively, resolved these problems by retaining a 10 percent tolerance for model year five. Table 4 provides a comparison of runs A2, A12, and A14.

Model run:	A2	A12	A14
Overmanning Tolerance			
Flying duties:	10%	10%	10%
Staff duties:	10%	10%	10%
Undermanning Tolerance			
Flying duties:	10%	* 5%	10%
Staff duties:	10%	10%	* 5%
(* = 10% for Year 5)			
Total Assignments For Years 1 thru 5			
To Flying duties:	4311	4467	4331
To Staff duties:	2199	2011	2163
Assignment Averaged Over Years 1 thru 5			
To Flying duties:	862	893	866
To Staff duties:	440	402	433

Table 4. Manning Tolerance vs. Assignment Distribution

As can be seen from Table 4, the reduced undermanning tolerance for flying duties resulted in more assignments to flying duties for model run A12. A similar effect did not occur with nonflying duties for model run A14, however. This result indicates that the original solution obtained in model run A2 tends to fill a greater percentage of nonflying duties than flying duties. This may indicate existence of alternate optimal solutions, since there are apparently more flying positions available than are being occupied. These results do, in general, show that the tolerances specified can affect the solution.

Attrition. The importance of attrition to the results of the GATES model is discussed in the section on tour lengths. Attrition also impacts the degree to which future manning requirements can be achieved. For example, the discussion of manning tolerances indicates that model year five required a greater undermanning tolerance than the other years. This appears to reflect the long term impact of attrition. The sensitivity of GATES to attrition rates was not directly tested. However, sufficient evidence has been presented to show that attrition is a very important factor in determining the final results.

Initial State of the System. The initial state of the system, which is determined by the input data, seems to have considerable effect on the model results. No actual sensitivity analysis was conducted (by using a different set of input data). However, examining the optimal solution results indicates that the initial system state may have prevented a zero-cost solution. The fact that all variations in input parameters that were performed resulted in largely the same number of gate failures indicates that these failures probably resulted from the initial state of the system. It is also worth noting that assignments leading to missed gates taper off in the later years of the modeled time horizon, indicating that the solution assignment flows provided by the GATES model result in improved gate compliance over time.

Practical Implications.

In general, the results of the sensitivity analysis performed on the GATES model indicate that the model has great potential utility in rated officer gate management. The model appears to be sensitive to those factors that seem most important in affecting officer rotations and gate compliance. Namely, the model is sensitive to tour lengths, attrition, and the initial state of the system. Additionally, overly restrictive manning tolerances are reflected in the GATES output file, "SANITY.OUT" (Appendix E) and result in aborted execution of the routine. Though the overall experience level specified for flying duties in this study seemed to have little effect on the model solution, the capability of specifying experience levels provides another means of achieving a degree of realism.

The impact of attrition--and the method for modeling attrition employed by GATES--seems to be the biggest obstacle to using the model efficiently. To avoid excessive distortion, care must be taken in performing any redistribution of attrition necessitated by an infeasible solution.

The next chapter contains suggestions for improvements to GATES that address the attrition modeling problem, as well as other model shortcomings.

V. Observations and Recommendations

This chapter discusses some of the general observations resulting from this study. Topics discussed include the adaptability of the GATES model, shortcomings and limitations of the model, and recommendations for further study.

Adaptability of the Model

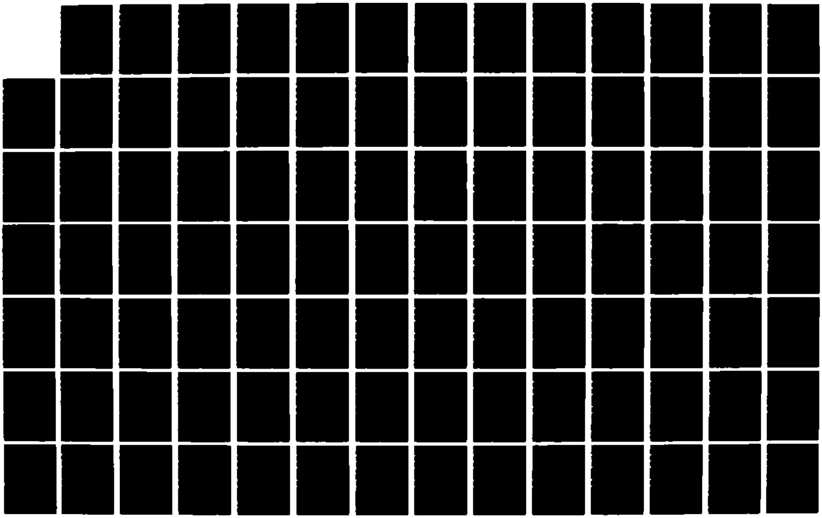
The adaptation and application of the GATES model to rated officer gate management will partially depend upon the degree to which model outputs agree with the real world rated officer assignment system. As presently formulated, GATES provides the means for specifying various parameters that can be used to increase the realism of the GATES solution. Through specification of these parameters, GATES should be capable of representing different weapon system groups and aeronautical ratings. Manning tolerances and constraints can be employed to model real world assignment constraints.

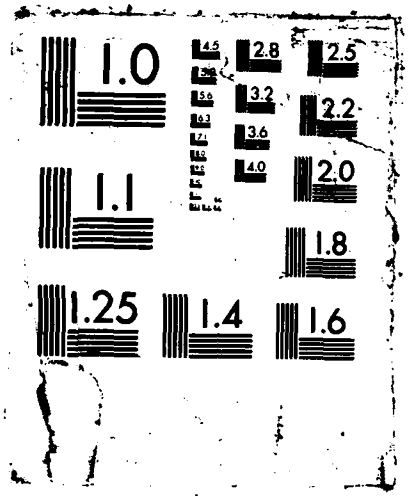
The solution provided by GATES theoretically represents the optimal assignment policy for minimizing flying gate noncompliance. The accuracy of the solution, of course, depends upon the accuracy of the input parameters. The potential existence of multiple optimal solutions provides some uncertainty in interpreting the results from GATES. However, application of additional manning constraints should tend to drive the solution towards a unique optimum.

The outputs from GATES (see Appendices E and F) provide information in various levels of detail. The year-by-year optimal assignment policies provide detail which probably exceeds the actual resolution of the model (an inherent characteristic of optimization methods). The summary data provided at the end of file "ROTEPLAN.OUT" (Appendix F) aggregates the solution and offers some insight into the overall gate management situation.

Basically, the output shows what is the best degree of gate achievement that can be expected (assuming the input parameters are realistic) if an assignment policy similar to that suggested by the output is followed. In a sense, this shows the critical bottlenecks in rated gate management. What is lacking, however, is identification of the nearly-critical bottlenecks. These can sometimes be discovered through sensitivity analysis by varying the constraint right hand sides (the manning requirements, manning tolerances, and experience level requirements). Finding alternate optimal solutions would help identify the ASD/ga[illegible] groups that bear watching by rated force manage[illegible] could be more readily accomplished if the pr[illegible] formulated as a general linear programm[illegible] techniques discussed by Steuer [illegible]

AD-A189 511 A NETWORK APPROACH TO RATED OFFICER GATE MANAGEMENT(U) 2/3
AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL
OF ENGINEERING M S OLSON DEC 87 AFIT/GOR/ENS/87D-13
UNCLASSIFIED F/G 5/9 NL




1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5

Besides gate information, the GATES model can be used to help identify problems with achieving desired manning levels. By inputting up-to-date attrition rates, GATES can be used to show the potential effects of attrition on future manning capabilities. The output data provided in file "SANITY.OUT" (Appendix E) is potentially useful in this regard.

Shortcomings and Limitations

The GATES network model offers the potential for significantly increased insight into the rated gate management problem. However, GATES has several shortcomings. Resolution of some of these shortcomings could improve the utility of the GATES model.

Level of Detail. One of the shortcomings of GATES pertains to the level of detail that it provides. The use of a minimum time increment of one year serves to hide some of the peaks and valleys in the assignment system, thereby partially defeating the initial impetus behind this research study.

Early in the development of GATES, an attempt was made to employ a half-year time increment. At that time, additional nodes and arcs with side constraints were used to model attrition (as opposed to the current implementation which uses nodal external flows to model attrition). The first computer run attempted with this higher resolution model was aborted after approximately six hours of real time and three hours of CPU time had elapsed (the computer system used was a Digital VAX 11/785). It was decided that a change back to the one-year time increment was necessary (at least temporarily) to permit rapid turn-around times for model development. Subsequent changes in the GATES model which have considerably reduced the number of nodes and arcs may permit a return to the half-year time increment without excessive run times. Use of a faster computer (such as the VAX 8650) would permit increased resolution without excessive run times. More efficient solution methods such as Karmarkar's linear programming algorithm would also permit increased resolution (15:75-90).

Attrition. The greatest shortcoming of GATES from a potential user's standpoint may be the difficulties that arise from the method employed to model attrition. By specifying the attrition demand for each node, the implicit assumption made is that there will be sufficient supply to each node to meet the attrition demand and to meet downstream flow requirements. Shortfalls in supply result in artificial variables in the solution set (infeasibility).

Early versions of GATES employed various means for dealing with attrition. The first method used was to artificially inflate manning requirements proportionate with the cumulative attrition amount. This method provided no means of differentiating between attrition rates for various ASD year groups.

Another method made use of attrition nodes and arcs. A separate node was used to model attrition for each time period and each ASD year group. A negative external flow was assigned to the node to represent the total attrition from the corresponding ASD year group for that time period. Upper bounds and side constraints on the arcs leading to the attrition nodes were used in an effort to distribute attrition as realistically as possible. The resulting network model was larger in terms of the number of nodes and arcs, and attrition seemed to occur in large blocks, instead of more uniformly for each particular ASD year group.

The current method for attrition modeling was chosen because of its structural simplicity and potential flexibility. However, the problem that accompanies this flexibility is the level of user involvement necessary. The greater potential for infeasible solutions also results in some difficulties with output interpretation.

An improved method of adjusting attrition to deal with infeasible problems would greatly improve the usability of GATES. It may be possible to develop an algorithm that compares the infeasible assignment solution, the artificial variables in the solution set, the nodal attrition demands, and the rotations into the network. This algorithm might then be able to specify the redistribution of attrition that appears most likely to result in feasibility, without overly distorting the results.

An alternative may be to employ a generalized network model which permits gains on the arcs. Such a model would permit assigning a multiplicative factor to each arc. This factor could represent the cumulative continuation rate for the ASD year group over the time duration represented by the arc. If this approach is taken, a more computer intensive solution technique such as the revised simplex linear programming method may be required (NETSID cannot handle arcs with gains). If the resulting computer run times are excessive, Karmarkar's algorithm may provide a viable solution method (15:75-90).

<u>Grouped Rotations</u>. Examining the output from GATES reveals that assignments tend to occur in large groups. That is, an entire ASD/gate credit group tends to be assigned to the same duty. This grouping seems to indicate that there may be alternate optimal solutions. Further investigation into the potential alternate optimality may improve the utility of GATES. Additional side constraints may also drive the solution to greater realism.

<u>Duty Durations</u>. The sensitivity analysis of the GATES model demonstrated the impact of duty tour length on an optimal assignment policy. The distribution, across model years, of rotations into the network and optimal solution assignments provides some insight into the approximate average duty duration.

Further investigations into duty tour lengths may provide the insight needed to determine the proper parameters to use for GATES. Examination of the input data and discussions with AFMPC suggest that there may even be different average tour lengths for different ASD year groups (35). A means of modeling any such differences would be a simple addition to GATES.

Because of the great interaction between duty tour lengths and attrition, a more effective means of dealing with infeasibilities resulting from excessive attrition demand could improve the adaptability of GATES to different duty durations. As an example of the impact of attrition on tour length modeling, the tour lengths had to be set at one year for the 18 year (plus) ASD group in order to supply sufficient flow to all ASD 18 nodes to meet the attrition demands due to retirements. Such a "fix" seems unrealistic--especially if applied to all ASD year groups.

Measuring Gate Attainment. The GATES model uses a simplified means for measuring flying gate achievement. Though AFR 36-20 specifies compliance with the second and third gates by the 18th year of aviation service, GATES provides no means of differentiating between the 18 year ASD group and groups that have passed the 18 year point (8:12). Because these groups are not separated, anyone who accomplishes nine (second gate) or eleven (third gate) years of flying is considered to have achieved the corresponding gate. Thus, GATES provides an avenue for subsequent gate compliance even after a gate has been missed. Though this does reflect a real world possibility, actual assignment considerations may prevent rotation of individuals in these older ASD year groups to flying duties.

The method for specifying gates is sufficiently flexible to allow adjustments or additions to the gate requirements. For example, specifying that the second and third gates must be achieved by 17 years of aviation service may be one means of arriving at a more realistic solution. Also, incremental degrees of gate achievement could be specified--such as requiring completion of 10 years of flying by 15 years of aviation service (that is, assigning a cost to arcs resulting in failures to meet this goal). Still, future efforts at continuing the work undertaken in this study might benefit from separating out some of the ASD groups currently grouped with ASD 18.

Information Format. The utility of the format of the information output by GATES is yet to be dete˜@ The detailed assignment solution provided in "ROTEPLAN.OUT" may be excessively detailed considering the resolution of the GATES model (see Appendix F). The summary tables provide a general picture of the optimal solution, but do not detail the particular ASD year groups that require close management. Perhaps an output which presents a compromise between these two extremes would be more useful and more representative of the capabilities of the GATES model.

Input Data Collection. Presently, GATES requires assembly of the input rotation data prior to model run time. This increases the user involvement required to run GATES and could result in considerable time expenditure--especially if GATES is to be run for all Air Force major weapon system groups and aeronautical ratings. An improved interface between GATES and the AFMPC database would probably increase the utility of the GATES model.

Recommendations

The shortcomings inherent in the present formulation of the GATES model suggest several model improvements. This section discusses these suggestions for further study.

Refinement. Though the verification and validation process accomplished in this study indicates that the GATES model is basically sound, there are some refinements to the model that may improve its application to rated gate management. These suggested refinements address many of the shortcomings of GATES identified above.

Increasing the level of detail of GATES by reducing the minimum time increment to a half-year period could provide improved realism in modeling average tour lengths. Additionally, a means for specifying different tour lengths for different ASD year groups may be beneficial.

Further investigation into real world assignment constraints may indicate the need for application of additional side constraints to the network model. Such side constraints may also reduce the tendency of grouped rotations in the optimal solution.

Identifying the means for determining the nature of alternate optimal or nearly optimal solutions to GATES could help identify the real gate management bottlenecks. Along similar lines, an output format that helps identify these "critical" ASD year groups may be beneficial to rated force managers.

A final recommendation for model refinement would be to extend the model time horizon beyond the current five year limit. Though the Rated Management Document provides detailed manning information for only the next five years, the numbers applicable to the fifth year could be extended out to the seven or eight year point (9). An extended time horizon may improve the realism of the optimal solution provided for the later years of the current five year horizon.

Revisions. Besides the model refinements identified above, some major revisions to the current GATES model may be appropriate. One suggested revision relates to the problems with attrition modeling. Another revision relates to the user interface.

The problems with the method currently used to model attrition have been discussed at length. There are at least two ways of dealing with this problem that could be examined. The first of these methods would involve an adaptation of the current attrition modeling method, but would provide a means of distributing attrition demands among the various nodes so as to minimize the chances of infeasibility. Much of the data that could be used to this end, such as rotation input data and nodal attrition demands, is already provided in the output file "SANITY.OUT" (Appendix E).

The second means of dealing with attrition would require application of a generalized network model with side constraints. This generalized network model would permit assigment of multiplicative factors (with values between zero and one) to the arcs. The factors would reflect the cumulative continuation rates for each ASD year group and should provide a more realistic distribution of attrition.

The final recommendation for model revision relates to the user interface. The current requirements to collect the input data prior to model run time and to enter input parameters individually into the GATES code require large expenditures of time. An improved interface, such as a spreadsheet program might provide, could increase the utility of GATES by simplifying its use. A decision support system approach such as that discussed by Sprague and Carlson may be appropriate (29). Development of a full decision support system based on GATES could result in a system usable directly by the AFMPC assignment officers.

Conclusion

The GATES rated officer management model, as currently implemented, offers a means for improved management of rated officers. Future studies aimed at addressing the shortcomings of GATES should provide additional benefits to personnel management, perhaps extending well beyond the narrow scope of this study.

Appendix A: GATES User's Guide

This guide contains basic instructions on running the computer programs that implement the GATES rated gate management model. This guide includes materials from the NETSID User's Guide by Kennington and Whisman that are specifically applicable to the GATES Model. For more detailed information on general use of the NETSID program, the user should refer to that document (Kennington & Whisman, 1987).

Computer System

The initial implementation of the GATES model was developed on a Digital Equipment Corporation VAX 11/785 computer running the VMS Version 4.5 operating system, and on a VAX 8650 running VMS Version 4.6. Although an attempt was made to apply standard Fortan 77 computer code, some modifications to the code may be necessary prior to running on other systems. The version of the NETSID network optimization routine provided by Dr. Kennington (apparently developed to operate on an IBM mainframe computer) was modified slightly as detailed below.

Computer Files

Descriptions of the program, data, and output files associated with the GATES model are provided in this section.

Program files. The program files, all written in Fortran 77, are as follows:

1. NETSID.for. This is the network optimization routine. For a more complete description, refer to the NETSID User's Guide. As used in this model, it produces two output files, "for007.dat" and "for008.dat."

2. Gates.for. This is the main control program used to implement the model. This program accomplishes the following tasks:

a. It provides a means for input of modeling parameters.

b. It reads the rotation data file, "rote.dat" (which must be build from the AFMPC data base before running the model).

c. It performs the computations necessary to describe the basic network structure of the model.

d. It prints information to the terminal screen and to an output file named "sanity.out" as a means of monitoring the performance of the model.

e. It builds the data files ("for001.dat" through "for004.dat") required as input by NETSID.

f. It calls the NETSID optimization routine.

g. It calls subroutine "netout". This subroutine reads the NETSID raw output file ("for008.dat") and converts it to information in the form of an "optimal" rotation policy (file "roteplan.out"). (The "netout" subroutine is included in the "gates.for" file.)

Input data file. A single input data file is required by the model. The "gates.for" program expects the data file to be named "rote.dat." The required Fortran format for each data line is (A4,2(I4),I3,8(I4)). The fields contain data as described here.

1. Current duty assignment (format A4).

a. "OPS" = flying duties, including advanced student duties and flying staff positions.

b. "SUP" = rated supplement and nonflying rated staff duties.

c. "AFIT" = Air Force Institute of Technology full-time graduate degree programs.

d. "PME" = resident professional military education (intermediate and senior service schools) and ASTRA (Air Staff Training) program.

2. Aviation service date (ASD) year group (format I4), in tenths of years, as of the start time of the model time horizon. Note that no decimal point should be used. Dividing the data file value by 10 will provide the number of years since ASD. Half-year increments provide more than sufficent detail for the model as currently implemented. As an example, a value of "105" would be used to represent 10-1/2 (10.5) years.

3. Flying gate credit accumulated (format I4), in tenths of years, as of the start time of the model time horizon. Again, no decimal point is used and half-year increments provide more than sufficient detail.

4. The remaining nine fields (format I3,8(I4)) contain integer values representing the number of individuals (with characteristics described by the first three fields) that arrived at their current duty station within specific time periods. The first of these fields represents the number of individuals who arrived over 4 years prior to the start time of the model. The next column represents the number that arrived 3-1/2 to 4 years prior to model start time. Successive columns represent successively more recent half-year increments. The last field represents the number of individuals who arrived on station within the 6-month period immediately preceding the model start time.

Output files. The main program ("gates.for") and subroutines ("NETSID.for" and "netout") produce several output files.

1. Sanity.out. This file is produced by the main program and is largely a regurgitation of the input data and calculations. The output format is designed to help verify input data and model performance.

2. Nodearc.out. This file is a listing of network arcs and associated beginning and ending nodes, along with the identifying characteristics of the nodes (time period, duty type, ASD year group, and flying gate

credit accumulation). Its purpose is to aid in troubleshooting model and program malfunctions.

3. For001.dat. This data file is output by the main program and is subsequently read by NETSID. It contains a list of node numbers with associated requirements (external flows). Nodes with a requirement of 0 may be omitted.

4. For002.dat. This data file is output by the main program and is subsequently read by NETSID. It contains a list of arc numbers, arc from-nodes, arc to-nodes, arc bounds, and arc costs. The rows must be ordered by arc number. An arc bound of -1.0 indicates no upper bound. (NETSID always assumes a lower bound of 0.)

5. For003.dat. This data file is output by the main program and is subsequently read by NETSID. It contains a list of side constraint numbers (identifying numbers), associated arc numbers, and a multiplier (usually 1.0). The rows must be ordered by arc number.

6. For004.dat. This data file is output by the main program and is subsequently read by NETSID. It contains a list of side constraint identifying numbers, the associated right-hand-side value, and an alpha identifier for the type of constraint (L for less-than-or-equal-to, E for equal-to, or G for greater-than-or-equal-to). The rows can be in any order.

7. For007.dat. This is the NETSID output file. It contains information about the network optimization procedure, the "optimal" objective function value, the number of iterations required to solve the problem, and a list of arc flows associated with the "optimal" solution.

8. For008.dat. This data file is output by NETSID and is subsequently read by subroutine "Netout." It contains only the arc flow information that is contained in "for007.dat."

9. Roteplan.out. This file contains the "optimal" assignment policy, as determined by the model. It is essentially a conversion of the arc flow values (contained in "for008.dat") into more meaningful information.

Using the Model

Using the GATES model involves four basic steps described below.

Building the input file. The input data file, "rote.dat," consists of the information described above. A separate program to search the AFMPC data base and output the required values was developed by AFMPC personnel to collect the data used in the initial implementation of the GATES model.

Parameter changes. An effort has been made to make the GATES model sufficiently flexible to handle most USAF major weapon system groups and aeronautical rating categories. The main program, "gates.for," provides the means for adjusting parameters such as manning requirements, average duty durations, and attrition rates. All such parameters are assigned in the second main section of the program (following the variable declarations and descriptions in section one). To change a parameter, one need only to edit the appropriate line(s) of the program file. Internal program documentation should be sufficient to direct a personnel analyst to the appropriate line(s).

Running the model. Once the input data file has been built and the model parameters have been adjusted, the actual running of the model is relatively simple. Begin by compiling "gates.for" (as well as "NETSID.for" if it has not previously been compiled). Then link these files. Finally, run the program. The VAX VMS commands for these steps are as follows:

```
fortran/list gates
link gates, netsid
```

run gates

Using these commands results in the program running interactively. When run interactively, "gates.for" (and NETSID) will provide some informational output to the terminal screen. Alternatively, the program can be run in batch by using a batch command file. During periods of heavy computer usage, the batch method may be preferable. Interactive run times for the initial implementation ranged from 10 minutes to over an hour, depending on the current level of computer system usage.

Output interpretation. Interpretation of the output from the GATES model must take into consideration the nature of a network optimization methodology. The "solution" provided is the "optimal" (least-cost) way of meeting the requirements (node demands) and side constraints. The cost of the objective function is the sum of all solution flows that are along arcs assigned a cost. The arcs assigned costs are those that lead to missed flying gates. The particular costs for each gate are specified in the input parameter section of the GATES code.

Multiple optimal solutions are likely to exist--especially when the optimal solution is achieved at zero cost. In some cases, multiple runs which impose successively more goals (potential contributors to the objective function) or side constraints (such as experience level requirements) may be required to determine the range of actions that could result in achievement of the initial goals.

Modifications to NETSID

A few modifications were made to the original version of NETSID provided by Dr. Kennington. They are listed here. The version provided to AFMPC/DPMYAF incorporates these changes.

1. NETSID was identified as a subroutine (instead of defaulting to a main program). The "return" command was also added.

2. Variable "QTEST" was set equal to 0 to turn off some print statements.

3. Two calls to "ERRSET" were commented-out (apparently IBM-specific and not recognized by VAX 11/785).

4. In subroutine "NSINPT," two print statements (to unit 7, i.e. file "for007.dat") were made conditional on QTEST *not being equal to zero*. This eliminated several lines of extraneous output to file 7.

5. In subroutine "REPORT," three write statements to unit 8 ("for008.dat")--which mirrored write statements to unit 7--were added to send raw arc flow data to "for008.dat."

6. All dimensions (parameters) were adjusted to the size of the Rated Gate Model.

A Note on Formats. During initial development of the GATES model, some apparent inconsistencies in data file formatting were observed. Provided here is a comparison of the NETSID format requirements as stated in the NETSID User's Guide and the corresponding read and write formats that were found to actually work. No changes were made to the format specifications contained in the NETSID routine.

1. For001.dat

User's guide: (I6,F10.2)
NETSID (read): (I6,F10.2,A1)
Gates (write): (' ',I5,F10.2)

2. For002.dat
User's guide: (3I6,2F10.2)
NETSID (read): (I6,3I6,2F10.0,4x,A10)
Gates (write): (' ',I5,2I6,2F10.2)

3. For003.dat
User's guide: (2I6,F10.0)
NETSID (read): (2I6,F10.0)
Gates (write): (' ',I5,I6,F10.0)

4. For004.dat
User's guide: (I6,F10.2,A1)
NETSID (read): (I6,F10.2,A1)
Gates (write): (' ',I5,F10.2,A1)

5. For008.dat
User's guide: n/a
Netout (read): (5x,I6,3(I8),2(E19.9))
NETSID (write): (5x,I6,3(2x,I6),2(2x,E17.9))

The only real conflicts between write and read formats appear to occur in the first field in data files 1 through 4. In all four cases, an "I6" in NETSID required a corresponding "I5" in "gates." Without this adjustment, the incorrect columns for each field were read in by NETSID.

Reference

Kennington, Jeffery L., and Whisman, Alan. NETSID User's Guide. Technical Report 86-OR-01. Department of Operations Research, Southern Methodist University, Dallas TX, June 1987.

Appendix B: GATES Program Listing

```
c***********************************************************************
c  MARK S. OLSON, AFIT thesis, DEC 87
c***********************************************************************
c
c   Main program to do the following:
c
c   1.  create NETSID data files for the flying gate management
c       problem,
c
c   2.  call NETSID network optimization routine, and
c
c   3.  call NETOUT subroutine to convert NETSID output data
c       to useable information.
c
c  The NETOUT subroutine is appended to the end of the GATES
c  program.  The NETSID optimization routine is in a separate
c  file.
c
c  NOTES:
c
c     (1) This program was developed around the Strategic
c  Airlift major weapon system group and the PILOT aeronautical
c  rating category.  During development, the goal was to build
c  sufficient flexibility into the program to allow it to be
c  adapted to other major weapon system groups and aeronautical
c  ratings with modification of only the input parameters
c  (variables) in Section II.  If that goal was not met, the
c  internal documentation will hopefully be sufficient to allow
c  an understanding of the program logic so that modifications
c  can be made.
c
c     (2) Section II of this program contains specifications for all
c  input parameters.  To modify a parameter, edit the appropriate
c  variable assignment statement in section II, then recompile this
c  program.
c
c     (3) Rotation input data for GATES should be in a file named
c  ROTE.DAT.
c
c     (4) To run GATES, it must be compiled and then linked with a
c  compiled version of the NETSID subroutine.
c
c     (5) GATES produces 7 output files:  SANITY.OUT, NODEARC.OUT,
c  FOR001.DAT, FOR002.DAT, FOR003.DAT, FOR004.DAT, and ROTEPLAN.OUT
c  (actually, ROTEPLAN.OUT is produced by the NETOUT subroutine).
c  Additionally, the NETSID optimization routine produces two files:
c  FOR007.DAT AND FOR008.DAT.
```

```
c
c     (6) Attrition is assumed to occur only at the time of expected
c  rotation.  The number of people attrited in each aviation service
c  date (ASD) year group is based on the number of people in the
c  corresponding ASD group at the beginning time and the product of
c  the appropriate attrition (continuation) rates for the attained
c  ASD group.  Attrition is divided among duty types and flying
c  accumulation (gate credit) values based on input information on
c  the distribution of each ASD year group by duty type (variable
c  DUTMAK) and gate credit values (variable FLYMAK).  An adjustment
c  in values of DUTMAK and FLYMAK may be necessary to achieve a
c  feasible solution.  Alternatively, variable CHANGE can be used to
c  adjust attrition at a particular node.
c
c     (7) Additional user information is in the GATES User's Manual.
c
c
c*******************************************************************
c
      program GATES
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c*******************************************************************
c*******************************************************************
cc                                                               cc
cc    SECTION I                                                  cc
cc                                                               cc
cc    VARIABLE DECLARATIONS AND DEFINITIONS                      cc
cc                                                               cc
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c   time = time period in fiscal years, 1 - 5
c   dutchr = (CHARACTER) type of duty assignment
c   duty = type of duty assignment, as follows:
c    duty = 1 = 'OPS' = flying duty
c    duty = 2 = 'SUP' = rated supplement & rated staff
c    duty = 3 = 'AFIT'
c    duty = 4 = 'PME' = PME & ASTRA
c   asdyg = aviation service date year group in years
c   asdten = asd in tenths of years (read from data file)
c   flyten = flying gate credit in tenths of years (from data file)
c   flycre = periods of flying gate credit accumulated in years
c   dur = temporary storage for duty assignment durations
c   minfly = minimum flycre value modeled for a particular ASD group
c   maxfly = maximum flycre value modeled for a particular ASD group
c   goals = number of gate credit requirements/policies/goals
c   nodnum = counter to assign sequential node numbers
c   maxnod = max number of nodes (but not the total number)
c   arcnum = counter to assign sequential arc numbers
c   begnod = beginning node assoc with a particular arc
c   endnod = ending node assoc with a particular arc
c   enddut = identifier for duty at end of an arc
```

```
c   newasd = identifier for asd group at end of an arc
c   oldasd = identifier for asd group at beginning of time horizon
c   newfly = identifier for flycre at end of an arc
c   endtim = ending time assoc with a particular arc
c   extflo = (REAL) external flow value for a particular node
c   arcost = cost assoc with flow on a particular arc
c   rcost = (REAL) cost assoc with flow on a particular arc
c   rbound = (REAL) upper bound on flow on a particular arc
c   constn = constraint number
c   ctype = constraint type
c    ctype = 0 = less than or equal to
c    ctype = 1 = greater than or equal to
c    ctype = 2 = equal to
c   ctypec = (CHARACTER) constraint type
c    ctypec = L = less than or equal to
c    ctypec = G = greater than or equal to
c    ctypec = E = equal to
c   rhs = (REAL) constraint right hand side
c   flag = flag variable used in sanity checks
c    flag = 0 = sanity checks passed
c    flag = 1 = sanity checks failed
c   flagl = flag used to determine which network 'nodes' should
c    be modeled and assigned a sequential network number.
c   expreq = number of experience level requirements (constraints)
c    specified in inputs section
c   pmetyp = breakout of 'PME' duty:  1=ASTRA,2=ISS,3=SSS
c   totin = (REAL) total of all inputs to the network (gains +
c    rotations)
c   totout = (REAL) total of all outflows from the network
c    (attrition)
c   makeup = (REAL) temporary storage for makeup of ASD year
c    groups by duty type and/or flying credit values
c
      integer time, duty, asdyg, flycre, asdten, flyten
      integer dur, minfly, maxfly, goals
      integer nodnum, maxnod, arcnum, begnod, endnod
      integer enddut, newasd, oldasd, newfly, endtim
      integer arcost, constn, ctype
      integer flag, flagl, expreq, pmetyp
c
      real extflo, rcost, rbound, rhs, totin, totout, makeup
c
      character ctypec
      character*4 dutchr
c*********************************************************************
c  the following variables are dimensioned for the number of ASD
c  YEAR GROUPS:
c
c  totasd(asdyg) = total of people in each asd group at initial
c   (start) time of model, based on 'rote.dat' file
c
      real totasd(0:18)
```

```
c*********************************************************************
c  the following are dimensioned for the number of TIME periods:
c
c  fy(time) = associated fiscal years
c  uft(time) = under-graduate flying training gains
c  faip(time) = FAIP/other gains
c  gains(time) = total number of UFT and FAIP/other gains
c  rote(time) = projected rotations into network
c  tadjov(time) = total overmanning adjustment
c  tadjun(time) = total undermanning adjustment
c  totflo(time) = total projected arc flows at particular time
c  asdgrp(time) = downstream asd group values
c  flyacc(time) = downstream flycre values
c  tinput(time) = total rotations into network incl gains
c  totatr(time) = total attrition count for year
c  atrtd(time) = total attrition to date
c  totreq(time) = total rqmts summed across duty types
c  ASTRA (time) = number of ASTRA requirements
c  ISS (time) = number of intermediate service school rqmts
c  SSS (time) = number of senior service school rqmts
c  comp18(time) = composite 1-year continuation rate for
c   asd year group 18+ (year groups 18 - 27)
c  catr18 (time) = cumulative attrition amount for asd group 18
c  aatr18 (time) = additional attrition for asd group 18 that
c   must be added due to those individuals who are above asd 18
c   (those who were in the initial asd group 18)
c  ccr18 (time) = cum cont rate for initial asd group 18
c  gtrote(time) = 'grand total rotations' = sum of all rotations
c   for year 'time'
c  gaintd (time) = gains to date
c  rotetd (time) = rotes to date
c  totund (time) = total undermanned requirement (across duties)
c  totove (time) = total overmanned requirement (across duties)
c  totadj (time) = total adjustment (sum across duty types)
C  FAILG1 (TIME) = TOTAL ROTATIONS INTO THE NETWORK AT 'TIME'
C   THAT HAVE FAILED TO MEET THE FIRST GATE
C  FAILG2 (TIME) = FAILED TO MEET SECOND GATE
C  FAILG3 (TIME) = FAILED TO MEET THIRD GATE
c
      integer fy(1:5), rote(1:5), asdgrp(1:5), flyacc(1:5)
      integer uft(1:5), faip(1:5)
      INTEGER FAILG1(1:5),FAILG2(1:5),FAILG3(1:5)
      real gains(1:5), tadjov(1:5), tadjun(1:5), totflo(1:5)
      real tinput(1:5), totatr(1:5), atrtd(1:5), totreq(1:5)
      real ASTRA(1:5), ISS(1:5), SSS(1:5)
      real catr18(0:5),aatr18(0:5),comp18(1:5),ccr18(1:5)
      real gtrote(1:5), gaintd(1:5)
      real rotetd(1:5),totund(1:5),totove(1:5),totadj(1:5)
c
c*********************************************************************
c  the following is dimensioned for the number of time periods in
c  the past that are used to group 'DATE-ARRIVED-STATION' blocks.
```

```
c
c  das(i) is the number of individuals that arrived on station at
c     'i' half-year periods prior to modeled time period 1.
c  das2(j) is used to combine half-year periods into full-year
c     periods.
c  sumtos(i)=total people that arrived on station in period i
c
      integer das(1:9), das2(1:5), sumtos(1:9)
c
c*****************************************************************
c  dimension for time periods in the past and duty types.
c
c  totdas(prior time period, duty) = 'total d.a.s.' by duty
c     type = sum of individuals by date-arrived-station and
c     duty type
c
      real totdas(1:9,1:4)
c
c*****************************************************************
c  dimension for time periods in the past, duties, and asd groups.
c
c  dasbya(prior time period, duty, asdyg) = 'D.A.S. by
c     asd group' = breakout of totdas by asd group.  Can be used to
c     determine if different asd groups seem to have different
c     duty durations.
c
      real dasbya(1:9,1:4,0:25)
c
c*****************************************************************
c  the following are dimensioned for the number of DUTY TYPES:
c
c  dutdur(duty) = duty duration in time periods
c  maxarc(duty) = the max arc number associated with assignments
c    to a particular duty ('duty'=5 is for attrition arcs)
c  avgtos(duty) = average time-on-station from AFMPC data file
c  sumdas(duty) = sum of all rotations (from d.a.s. data) for
c    each duty
c  gtdut(duty) = 'grand total duty' = grand total sum of all
c    people initially in each duty type
c
      integer dutdur(1:4), maxarc(0:5)
      real avgtos(1:4), sumdas(1:4), gtdut(1:4)
c
c******************************************************************
c  the following are dimensioned for TIME periods and DUTY TYPES:
c
c  rqmt(time,duty) = manning rqmt from rated mgt document
c  totdut(time,duty) = total rotations into the network
c    from 'duty' at 'time'
c  adjust(time,duty) = people holding positions but not
c    yet in the network (first rotation is downstream)
c  adjove(time,duty) = adjusted overmanned requirement
```

```
c  adjund(time,duty) = adjusted undermanned requirement
c  reqove(time,duty) = max overmanned requirement
c  reqund(time,duty) = min undermanned requirement
c  pctove(time,duty) = percent overmanning allowed
c  pctund(time,duty) = percent undermanning allowed
c
      real rqmt(1:5,1:4), totdut(1:5,1:4)
      real adjust(1:5,1:4), adjove(1:5,1:4), adjund(1:5,1:4)
      real reqove(1:5,1:4), reqund(1:5,1:4)
      real pctove(1:5,1:4), pctund(1:5,1:4)
c*******************************************************************
c  the following are dimensioned for TIME periods and ASD groups:
c
c  conasd(time,attained asd group) = 1-year continuation rate
c  ccrasd(time,attained asd group) = cumulative continuation rate
c   to date
c  atrit(time,attained asd group) = number of attritions
c  catrit(time,attained asd group) = cumulative attritions to date
c  ccrfa(time,attained asd group) = cumulative continuation rate
c   for FAIP/other gains
c  catrfa(time,attained asd group) = cum number of attritions
c   for FAIP/others to date
c  atrfa(time,attained asd group) = number of attritions for FAIPs
c  atrflo(time,attained asd group) = sum of all attritions for
c   each asd year group at 'time'
c  rotbya(time, initial asd group) = number of rotations into
c   the network by asd year group
c
      real conasd(1:5,0:18), ccrasd(1:5,0:18), atrit(1:5,0:18)
      real catrit(0:5,0:18), ccrfa(1:5,4:8), catrfa(1:5,4:8)
      real atrfa(1:5,4:8), atrflo(1:5,0:18), rotbya(1:5,0:18)
c
c********************************************************************
c  dimension for ASD YEAR GROUPS and FLYcre:
c
c   flytot(asdyg,flycre) = initial makeup of each asd group
c    by flycre values (raw numbers from AFMPC rotation data)
c   flypct(asdyg,flycre) = initial makeup of each asd group
c    by flycre values (percent); calculated from raw input
c    rotation data
c   flymak(asdyg,flycre) = approx makeup of each asd group
c    by flycre values (percent); input for use in calculating
c    attrition
      real flypct(0:18,0:11), flytot(0:18,0:11)
      real flymak(0:18,0:11)
c********************************************************************
c  dimension for DUTY and ASD YEAR GROUPS
c
c  asdpct(duty,asdyg) = initial makeup of each duty type
c   by asd year group values (percent); calculated from raw
c   input rotation data
c  asdsum(duty,asdyg) = sum of people in each duty type/
```

```
c   asd year group combination
c  dutpct(duty,asdyg) = initial makeup of each asd year group
c   by initial duty assignment occupied (percent); calculated
c   from raw input data
c  dutmak(duty,asdyg) = distribution of each asd year group
c   among duty types (percent); input for use in calculating
c   attrition
c
      real asdpct(1:4,0:18), asdsum(1:4,0:18), dutpct(1:4,0:18)
      real dutmak(1:4,1:18)
c
c*********************************************************************
c  the following are dimensioned for TIME periods, DUTY TYPES (or
c  PME types), and CONSTRAINT TYPES:
c
c  constr(time,duty,constraint type) = associated constraint number
c  pmecon(time,pme type,constraint type) = assoc constraint number
c
      integer constr(1:5,1:4,0:2), pmecon(1:5,1:3,0:1)
c*********************************************************************
c  the following is dimensioned for TIME periods, DUTY TYPES, and
c  ASD year groups:
c
c  totrot(time,duty,asdyg) = total number of rotations
c   into network based on INITIAL asd group and duty type
c
      real totrot(1:5,1:4,0:18)
c*********************************************************************
c  dimension for time periods, duty types, asd groups, flycre values:
c
c  atr(time,duty,asdyg,flycre) = attrition at the
c    corresponding node
C  CHANGE(TIME,DUTY,ASD YEAR GROUP,FLY CREDIT) = ADJUSTMENT IN
C    ATTRITION AT CORRESPONDING NODE;
C    DUTY TYPE CAN ONLY BE 1 OR 2, MIN ASD AND GATE CREDIT IS
C    6 BECAUSE ATTRITION IS NEGLIGIBLE FOR VALUES BELOW THIS.
c
      real atr(1:5,1:4,0:18,0:11),CHANGE(1:5,1:2,6:18,6:11)
c*********************************************************************
c  the following are dimensioned for the total number of gate 'GOALS':
c
c  cost(goal number) = artificially assigned cost for not attaining
c   associated flying 'gate'
c  asd(goal number) = aviation service date assoc with 'gate' goal
c  fly(goal number) = flying time credit assoc with 'gate' goal
c
      integer cost(1:12), asd(1:12), fly(1:12)
c*********************************************************************
c  dimension for the number of experience level requirements specified
c  in the inputs (value of EXPREQ)
c  expcon must be dimensioned for value of expreq and time
c
```

```
      integer expdut(5),expasd(5),exptyp(5),expcon(5,1:5)
      real exppct(5)
c*********************************************************************
c  ARC is dimensioned for the MAX NUMBER OF ARCS.
c  Arc(arcnum,index):
c   index=1 holds 'from node'
c   index=2 holds 'to node'
c   index=3 holds 'arc cost'
c   index=4 holds 'arc flow upper bound'
c   index=5 holds 'arc begin time'
c   index=6 holds 'arc end time'
c   index=7 holds 'arc begin asd year group'
c   index=8 holds 'arc end asd year group'
c   INDEX=9 HOLDS 'ARC BEGIN FLY CREDIT'
C   INDEX=10 HOLDS 'ARC END FLY CREDIT'
C
      integer arc(3000,1:10)
c*********************************************************************
c  NODE is dimensioned for the number of TIME periods, DUTY TYPES,
c   ASD year groups, and FLY CREDIT values
c   (the 'FULL' size of the network).
c  Node(time,duty,asdyg,flycre,index):
c   index=0 holds 'node number'
c   index=1 holds 'node external flow'
c
      integer node(1:5,1:4,0:18,0:11,0:1)
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c*********************************************************************
c*********************************************************************
cc                                                                 cc
cc    SECTION II                                                   cc
cc                                                                 cc
cc    SPECIFY PARAMETERS                                           cc
cc                                                                 cc
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  **** This info could probably be input from
c  keyboard when prompted by the program (interactive input)
c  with minor modifications to this program. ****
c********************************+************************************
c  starting fiscal year for the network model is fy(1).
c
      fy(1)=88
      fy(2)=fy(1)+1
      fy(3)=fy(1)+2
      fy(4)=fy(1)+3
      fy(5)=fy(1)+4
c*********************************************************************
c  duty assignment average durations by (duty) in years
c
      dutdur(1)=3
```

```
      dutdur(2)=3
      dutdur(3)=1
      dutdur(4)=1
c*********************************************+****************************
c  artificial cost(i) assigned for failure to meet fly(i) years of
c  gate credit accumulation by asd(i) years of aviation service.
c  These costs drive the network optimization routine to provide a
c  solution which maximizes compliance with gate rqmts and goals.
c  NOTE:  Cost values need to be relatively small integer values
c  (less than 10 ?) to prevent overflow of the objective function
c  in the NETSID routine.
c  ****
c  Costs must be ordered high-to-low, unless it can be determined
c  that no other costs apply to a particular arc.  This is because no
c  additional costs are examined once it is determined that a particular
c  cost applies to an arc (costs are examined sequentially based on
c  the 'index' i.  However, zero-costs can be
c  interspersed with the other values (this prevents a need to
c  constantly renumber the 'i's associated with costed arcs).
c  ****
c  'Goals' holds the number of requirements/policies/goals.
c  If more than 12, must redimension cost(i), fly(i), asd(i)
c  (INTEGERS !!)
c
c 'requirements'
c   first gate:
      cost(1)=3
      fly(1)=6
      asd(1)=12
c   second gate:
      cost(2)=2
      fly(2)=9
      asd(2)=18
c   third gate:
      cost(3)=1
      fly(3)=11
      asd(3)=18
c   ?? gate:
      cost(4)=0
      fly(4)=0
      asd(4)=0
c   ?? gate:
      cost(5)=0
      fly(5)=0
      asd(5)=0
c
c 'policies' for years thru asd year 6 are necessary to
c  prevent large flows (towards the end of the network time horizon)
c  to nonfly jobs for these low-time flyers.
c  (NOTE:  these are NOT required for this latest version of GATES
c  due to internal structural constraints which force low time
c  aviators to rotate only to flying duties.
```

```
c
      cost(6)=4
      fly(6)=1
      asd(6)=1
c
      cost(7)=4
      fly(7)=2
      asd(7)=2
c
      cost(8)=4
      fly(8)=3
      asd(8)=3
c
      cost(9)=4
      fly(9)=4
      asd(9)=4
c
      cost(10)=4
      fly(10)=5
      asd(10)=5
c
      cost(11)=4
      fly(11)=6
      asd(11)=6
c
c   'goals'
c    not used:  suggest assigning cost of '1' if used
c
      cost(12)=0
      fly(12)=0
      asd(12)=0
c
c   total number of requirements+policies+goals
c
      goals=12
c
c**********************************************************************
c   projected UFT gains: assumption is entry at 1 year asd group
c   and 1 year fly credit.  Fiscal year totals from the rated management
c   document are used.
c   It is assumed that these values are already adjusted for attrition.
c   (INTEGERS !!!)
c
c   1st year
      uft(1)=229
c   2nd year
      uft(2)=248
c   3rd year
      uft(3)=248
c   4th year
      uft(4)=235
c   5th year
```

```
      uft(5)=235
c
c  projected FAIP/other gains:  assumption is entry at 4 year
c  asd group and 4 year fly credit.  Fiscal year quantities from
c  the rated management document are used.  It is assumed that these
c  values are adjusted for attrition.
c  (INTEGERS !!!)
c
c  1st year
      faip(1)=88
c  2nd year
      faip(2)=67
c  3rd year
      faip(3)=67
c  4th year
      faip(4)=64
c  5th year
      faip(5)=64
c
c  total UFT and FAIP/OTHER gains for use in 'sanity' checks.
c
      do 10 time=1,5
        gains(time)=uft(time)+faip(time)
        node(time,1,1,1,1)=uft(time)
        node(time,1,4,4,1)=faip(time)
        do 10 i=1,time
          gaintd(time)=gaintd(time)+gains(i)
10    continue
c**********************************************************************
c  requirements for fiscal years of network time horizon for use
c  as side constraints:  Rqmt(time,duty).
c  (REAL NUMBERS !!!)
c
c  flying positions
c  NOTE:  THIS IS THE SUM OF 'FORCE' + 'TRAINING' + 'ADVANCED
c  STUDENT' + 'ATC MWS PRESENCE' FROM THE RATED MANAGEMENT DOC.
c
      rqmt(1,1)= 2201.
      rqmt(2,1)= 2195.
      rqmt(3,1)= 2192.
      rqmt(4,1)= 2159.
      rqmt(5,1)= 2191.
c
c  rated staff/supplement positions (nonflying)
c  NOTE:  THIS IS THE SUM OF 'STAFF' + 'GENERAL OPS STAFF' +
C  'SUPPLEMENT' FROM THE RATED MANAGEMENT DOCUMENT
C
      rqmt(1,2)= 979.
      rqmt(2,2)= 977.
      rqmt(3,2)= 1041.
      rqmt(4,2)= 1013.
      rqmt(5,2)= 1192.
```

```
c
c  ********************************
c  NOTE:  the Rated Management Document groups AFIT and PME together.
c  In this model, the requirements have been split 50-50 between AFIT
c  and PME/ASTRA.
c  **** Better information would be 'nice' to have. ****
c  ********************************
c
c  AFIT positions (nonflying)
      rqmt(1,3)= 31.
      rqmt(2,3)= 31.
      rqmt(3,3)= 33.
      rqmt(4,3)= 32.
      rqmt(5,3)= 38.
c
c  PME/ASTRA positions (nonflying)
c
      rqmt(1,4)= 31.
      rqmt(2,4)= 31.
      rqmt(3,4)= 34.
      rqmt(4,4)= 33.
      rqmt(5,4)= 38.
c
c  breakout of PME/ASTSA positions.  This is to insure adequate
c  flows to 'PME' nodes for each range of ASD year groups eligible
c  to attend one of these assignments.  ASTRA(time) + ISS(time) +
c  SSS(time) should equal rqmt(time,4).  This set of inputs
c  assumes 5 ASTRA positions each year and the remaining 'PME'
c  positions are split 70 % to ISS and 30% to SSS.  These numbers
c  can be easily adjusted in this section.
c
      do 11 time=1,5
        ASTRA(time) = 5.
        ISS(time) = ANINT(.7 * (rqmt(time,4)-ASTRA(time)))
        SSS(time) = rqmt(time,4)-ASTRA(time)-ISS(time)
11    continue
c
c  sum requirements across duty types
c
      do 12 time=1,5
        do 12 duty=1,4
          totreq(time)=totreq(time) + rqmt(time,duty)
12    continue
c*****************************************************************
c  input approximate experience level requirements by duty types.
c  This is more likely needed for FLYING duties to prevent low
c  overall experience levels in those duties.
c  ***
c  Note:  For less-than constraints, beginning asd groups determine
c  which flows apply against the constraints.  For greater-than
c  constraints, even if specified asd group is attained only at the
c  end of the assignment, the flow contributes towards attainment
```

```
c  of the goal.
c  ***
c  Exptyp(i) = 0 for less-than constraints
c            = 1 for greater-than constraints
c  Exppct(i) is applied against the baseline manning requirement
c    for the corresponding duty (not adjusted for over-/under-
c    manning tolerances).
c
c  CONSTRAINT SUGGESTED BY DPMYAF:
c  At least 50 percent of duty type 1 positions must be held by
c  personnel holding an asd group of 6 or higher:
      expdut(1)=1
      expasd(1)=6
      exppct(1)=50.
      exptyp(1)=1
c
c  Additional experience level requirements could be specified using
c  expdut(2),....., expdut(3), etc.
c
c  expreq is the total number of experience level requirements
c  identified above:
      expreq=1
c*********************************************************************
c  input of undermanning and overmanning tolerances for calculation of
c  side constraints and negative external flows at end of time horizon.
c  Percent overmanning and undermanning allowable (deviations from
c  'Rqmt' values input above) by time and duty type.
c  NOTE:  For this set of inputs, a do-loop is used to assign the same
c  tolerance percentages for all years.  However, a very minor change
c  will allow using different percentages for each year.
c
c  **** May need to adjust these to get a feasible solution. ****
c
      do 16 time=1,5
        pctove(time,1) = 10.
        pctove(time,2) = 10.
        pctove(time,3) = 5.
        pctove(time,4) = 0.
c
        pctund(time,1) = 10.
        pctund(time,2) = 10.
        pctund(time,3) = 5.
        pctund(time,4) = 0.
16    continue
c
c*********************************************************************
c  Input continuation rates (for this MWS and rating
c  Category) by year and asd group.  Continuation rates for asd group
c  18 years is weighted average of rates for years 18 up to 21,
c  depending on the 'time'.  This 18-year rate applies only to those
c  entering asd group 18+ during the time horizon of the model.  Those
c  individuals who start in asd year group 18, and therefore stay in asd
```

```
c  group 18 during the model, make use of 'comp18', which is an averaged
c  continuation rate depending on the attained asd year groups of the
c  initial 18-year group.
c
c  **** For this set of inputs, a do-loop is used to assign the same
c  continuation rate for all 5 years modeled.  However, separate
c  rates for each year could be input. ****
c
      do 18 time=1,5
        conasd(time,0)=1.0
        conasd(time,1)=1.0
        conasd(time,2)=1.0
        conasd(time,3)=1.0
        conasd(time,4)=1.0
        conasd(time,5)=1.0
        conasd(time,6)=.94
        conasd(time,7)=.83
        conasd(time,8)=.83
        conasd(time,9)=.86
        conasd(time,10)=.86
        conasd(time,11)=.86
        conasd(time,12)=.93
        conasd(time,13)=.98
        conasd(time,14)=.99
        conasd(time,15)=.99
        conasd(time,16)=.99
        conasd(time,17)=.99
18    continue
c
c  conasd rates for asd year group 18(+) are weighted averages for
c  the asd groups that makeup 'asd year group 18'.  The makeup of
c  this group changes as time progresses.
c
c  continuation rate for year 1, asd group 18+ is set equal to 1.0
c  because attrition for this group is calculated using rates for
c  asd 18+ provided below (comp18).
      conasd(1,18)=1.0
c  year 2 rate for year group 18
      conasd(2,18)=.99
c  year 3 weighted average for asd groups 18-19
      conasd(3,18)=.97
c  year 4 weighted average for asd groups 18-20
      conasd(4,18)=.88
c  year 5 weighted average for asd groups 18-21
      conasd(5,18)=.84
c
c  calculate cumulative continuation rates.  These are indexed on
c  the current 'time' and the attained asd group at 'time'.
c  Cumulative continuation rates for FAIP gains and the initial
c  group of individuals in asd group 18+ are calculated separately.
c
      do 20 newasd=1,18
```

```
         do 20 time=1,5
           ccrasd(time,newasd)=conasd(time,newasd)
           do 20 i=1,time-1
             oldasd=newasd-i
             if (oldasd.lt.0) oldasd=0
             ccrasd(time,newasd)=ccrasd(time,newasd) *
     $          conasd(time-i,oldasd)
20     continue
c
c  composite continuation rates for the individuals originally in
c  asd group 18 are a weighted average for the 'attained' asd groups
c  for this group as of 'time'.  comp18(1) is the 'average' of
c  rates for year groups 18-27, ..., comp18(5) is the 'average'
c  for year groups 22-27.
c
       comp18(1)=.83
       comp18(2)=.55
       comp18(3)=.30
       comp18(4)=.17
       comp18(5)=.14
c
c  cumulative continuation rates for the original asd year group 18.
c
       do 22 time=1,5
         ccr18(time)=1.0
         do 22 i=1,time
           ccr18(time)=ccr18(time) * comp18(i)
22     continue
c
c*********************************************************************
c  input asd group distribution percentages (approx) for calculation
c  of attrition negative external flows.
c
c  NOTE:  Only asd groups GE 6 are included here (assumption is that
c    attrition is negligible for asd groups LT 6)
c
c  flymak(attained asd group, attained fly credit) = 'flycre makeup'
c    = percent of each asd group consisting of each flycre value.
c    For each asd group, summing across the flycre values should
c    equal 100.
c  CHANGING VALUES OF FLYMAK AND DUTMAK (BELOW) DIRECTLY AFFECTS
c  HOW MANY INDIVIDUALS FROM EACH DUTY,ASD YEAR GROUP,FLY CREDIT
c  COMBINATION ARE ATTRITED EACH YEAR.
c
       flymak(6,6)=100.
c
       flymak(7,7)=100.
c
       flymak(8,8)=100.
c
c       flymak(9,6)=20.
c       flymak(9,7)=20.
```

```
c        flymak(9,8)=30.
c        flymak(9,9)=30.
       flymak(9,9)=100.
c
c        flymak(10,6)=10.
       flymak(10,7)=15.
c        flymak(10,8)=15.
c        flymak(10,9)=30.
c        flymak(10,10)=30.
       flymak(10,9)=15.
       flymak(10,10)=70.
c
       flymak(11,6)=10.
       flymak(11,7)=10.
       flymak(11,8)20.
c        flymak(11,9)=20.
c        flymak(11,10)=20.
c        flymak(11,11)=20.
       flymak(11,9)=10.
       flymak(11,10)=25.
       flymak(11,11)=25.
c
       flymak(12,7)=10.
       flymak(12,8)=10.
       flymak(12,9)=15.
       flymak(12,10)=15.
       flymak(12,11)=50.
c
       flymak(13,8)=15.
       flymak(13,9)=15.
       flymak(13,10)=15.
       flymak(13,11)=55.
c
       flymak(14,9)=20.
       flymak(14,10)=20.
       flymak(14,11)=60.
c
       flymak(15,9)=15.
       flymak(15,10)=20.
       flymak(15,11)=65.
c
       flymak(16,9)=15.
       flymak(16,10)=15.
       flymak(16,11)=70.
c
       flymak(17,9)=15.
       flymak(17,10)=15.
       flymak(17,11)=70.
c
c        flymak(18,9)=10.
c        flymak(18,10)=10.
c        flymak(18,11)=80.
```

```
      flymak(18,11)=100.
c
c  Specify the distribution of each asd group among duties, for use
c  in attrition.  DUTMAK(duty,asd group) = 'duty makeup' =
c  the percent of each asd group that is assigned to the indicated
c  duty type.  For each asd group, summing across duty types should
c  total 100 percent.  ASSUMPTION: attrition at AFIT and PME nodes
c  is zero.  Therefore, the percentages really refer only to the
c  approximate relative percentages of each asd group occupying duty
c  types 1 (ops fly) and 2 (supplement/staff).
c  CHANGING THESE VALUES, ALONG WITH CHANGING VALUES OF FLYMAK (ABOVE)
c  DIRECTLY AFFECTS ATTRITION FOR EACH DUTY,ASD GROUP,FLY CREDIT
c  COMBINATION.
c
      dutmak(1,6)=100.
      dutmak(1,7)=100.
      dutmak(1,8)=100.
c
c      dutmak(1,9)=75.
c      dutmak(2,9)=25.
      dutmak(1,9)=100.
c
      dutmak(1,10)=50.
      dutmak(2,10)=50.
c
      dutmak(1,11)=50.
      dutmak(2,11)=50.
c
      dutmak(1,12)=50.
      dutmak(2,12)=50.
c
      dutmak(1,13)=60.
      dutmak(2,13)=40.
c
      dutmak(1,14)=60.
      dutmak(2,14)=40.
c
      dutmak(1,15)=60.
      dutmak(2,15)=40.
c
      dutmak(1,16)=60.
      dutmak(2,16)=40.
c
      dutmak(1,17)=60.
      dutmak(2,17)=40.
c
      dutmak(1,18)=60.
      dutmak(2,18)=40.
C*********************************************************************
C  USE PARAMETER 'CHANGE' TO ADJUST THE ATTRITION AT A PARTICULAR
C  NODE IN ORDER TO ACHIEVE A FEASIBLE SOLUTION.  + INCREASES THE
C  AMOUNT OF ATTRITION, - DECREASES ATTRITION.
```

```
C  CHANGE(TIME,DUTY,ASD GROUP,GATE CREDIT)
C
      CHANGE(1,1,9,9)=-3.
      CHANGE(1,1,9,8)=3.
      CHANGE(1,2,12,8)=-1.
      CHANGE(1,1,12,12)-1.
      CHANGE(2,1,8,8)=-5.
      CHANGE(2,1,8,7)=5.
      CHANGE(2,1,9,9)=-5.
      CHANGE(2,1,9,8)=5.
      CHANGE(3,1,9,9)=-8.
      CHANGE(3,1,9,8)=8.
      CHANGE(3,1,12,10)=-1.
      CHANGE(3,1,12,11)=1.
      CHANGE(5,1,8,8)=-40.
      CHANGE(4,1,7,7)=40.
      CHANGE(5,1,9,9)=-2.
      CHANGE(4,1,8,8)=2.
      CHANGE(5,1,11,11)=-4.
      CHANGE(4,1,10,10)=4.
C
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c***********************************************************************
c****************************************+******************************
cc                                                                    cc
cc   SECTION III                                                      cc
cc                                                                    cc
cc   INPUT ROTATION DATA (POSITIVE EXTERNAL FLOWS)                    cc
cc                                                                    cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  this information comes from a datafile created from the AFMPC
c  database.  Given the correct database access commands, THIS program
c  could possibly CREATE the required database, using parameters
c  specified via interactive inputs to this program.  As presently
c  implemented, though, the data file must be created before running
c  this program (named 'rote.dat').
c
      open (unit=5,file='rote.dat',status='old')
c*******************************************************+***************
c  ********** begining of goto loop to read data **********
c
35    continue
c
c  initialize das and rote
c
      do 30 i=1,9
        das(i)=0
30    continue
c
      do 32 time=1,5
        rote(time)=0
```

```
32     continue
c
c  read rotation data file
c
       read (5,300,end=40), dutchr,asdten,flyten,das(9),
      $ das(8),das(7),das(6),das(5),das(4),das(3),das(2),das(1)
300    format (A4,I4,I4,I3,8(I4))
c**********************************************************************
c  convert duty type from character to integer
c
       if (dutchr.EQ.'OPS') duty=1
       if (dutchr.EQ.'SUP') duty=2
       if (dutchr.EQ.'AFIT') duty=3
       if (dutchr.EQ.'PME') duty=4
c*************************************************+****************
c  convert das(i) to rote(j), that is, how many individuals will
c  rotate in time period j.
c  das2(k) is used to total half-year
c  date-arrived-station (d.a.s.) groups into full-year groups.
c
       das2(1)=das(1)+das(2)
       das2(2)=das(3)+das(4)
       das2(3)=das(5)+das(6)
       das2(4)=das(7)+das(8)
       das2(5)=das(9)
c
       dur = dutdur(duty)
c
c  if the number of time periods since date arrived station (d.a.s.)
c  is greater than or equal to the modeled duration for the
c  particular duty type, then the people are assumed to rotate
c  in time period 1.  Otherwise, they are assumed to rotate 'dutdur'
c  time periods after 'd.a.s.'.
c
       do 34 i=dur,5
         rote(1)=rote(1) + das2(i)
34     continue
       do 36 time=2,dur
         rote(time) = das2(dur+1-time)
36     continue
c************+*******************************************************
c  convert asd year group from tenths-of-years to FULL-years
c  ********** (INTEGER DIVISION) *********
       asdyg = asdten  / 10
c
c  convert fly credit from tenths-of-years to FULL-years
c  ********** (INTEGER DIVISION) *********
       flycre = flyten  / 10
c*********************************************************************
c  accumulate date-arrived-station figures.
c
       do 37 i=1,9
```

```
        totdas(i,duty)=totdas(i,duty)+das(i)
        dasbya(i,duty,asdyg)=dasbya(i,duty,asdyg)+
     $    das(i)
37    continue
c**********************************************************************
c  accumulate total rotations by time and duty type and asd group
c  (based on INITIAL ASD GROUP).
c  These are used to adjust the manning requirements
c  that are used as side constraints.
c  These values are also used in the 'sanity' checks, and to
c  compute attrition quantities.
c  Note that these numbers do not include UFT and FAIP/other gains.
c
      if (asdyg.gt.18) asdyg=18
      do 38 time=1,5
        gtrote(time)=gtrote(time)+rote(time)
        totrot(time,duty,asdyg) =
     $    totrot(time,duty,asdyg)+rote(time)
c
38    continue
c**********************************************************************
c  adjust downstream asd year group
c
      asdgrp(1)=asdyg
      do 39 time=2,5
        asdgrp(time)=asdgrp(time-1) + 1
39    continue
c
c  force max asd year group to be 18 years.
c  This is a modeling consideration to reduce the size of the network
c
      do 41 time=1,5
        if (asdgrp(time).GT.18) then
          asdgrp(time)=18
        endif
41    continue
c**********************************************************************
c  Force min flycre to be asd group or 6 years, whichever
c  is less.
c  Force all rotations for asd group less than 6 years
c  to be from fly jobs.
c  Force individuals currently in AFIT positions with ASD group
c  greater than 13 to enter the network at duty type 2 (SUP),
c  because AFIT nodes do not exist in the network for these ASDs.
c  Force individuals currently in PME positions with ASD groups
c  between 8 and 13 or between 15 and 18 to enter the network at
c  duty type 2 also.
c  NOTE that these are modeling assumptions which reduce the size
c  of the network.  Affected individuals are totalled against
c  actual current duty in the above loop (totrot) for use in
c  adjusting the 'requirements' side constraints.
c
```

```
      if (asdyg.GE.6.AND.flycre.LT.6) then
        flycre=6
      endif
c
      if (asdyg.LT.6) then
        flycre=asdyg
        duty=1
      endif
c
      if (duty.eq.3.and.asdyg.gt.13) duty=2
c
      if (duty.eq.4) then
        if ((asdyg.lt.7).or.
     $     (asdyg.gt.8.and.asdyg.lt.13).or.
     $     (asdyg.gt.15.and.asdyg.lt.18)) duty=2
      endif
c
c  adjust downstream accumulated fly credit for flying duties
c
      do 42 time=1,5
        if (duty.EQ.1) then
          flyacc(time)=flycre+time-1
        else
          flyacc(time)=flycre
        endif
42    continue
c
c  force max fly credit to be 11 years.  This is a modeling
c  consideration to reduce size of the network.  11 years
c  corresponds to completion of the third (and last) flying 'gate'.
c
      do 44 time=1,5
        if (flyacc(time).GT.11) then
          flyacc(time)=11
        endif
44    continue
c**********************************************************************
c  total initial personnel by asd year group/fly credit
c
      do 45 time=1,5
        flytot(asdyg,flycre)=flytot(asdyg,flycre) +
     $    rote(time)
45    continue
C**********************************************************************
C  TOTAL INDIVIDUALS WHO WILL ALREADY HAVE FAILED TO MEET A GATE AT
C  THE TIME OF THEIR FIRST ROTATION INTO THE NETWORK
C
      DO 43 I=1,5
        IF (FLYACC(I).LT.6.AND.ASDGRP(I).GE.12)
     $    FAILG1(I)=FAILG1(I)+ROTE(I)
        IF (FLYACC(I).GE.6.AND.FLYACC(I).LT.9.AND.
     $    ASDGRP(I).GE.18)
```

```
     $     FAILG2(I)=FAILG2(I)+ROTE(I)
        IF (FLYACC(I).GE.9.AND.FLYACC(I).LT.11.AND.
     $     ASDGRP(I).GE.18)
     $     FAILG3(I)=FAILG3(I)+ROTE(I)
43    CONTINUE
c**********************************************************************
c  place external flow values into node array.  Values are added to
c  existing entries because above conversions of flycre and duty type
c  could result in more than one line of the data file supplying values
c  to a single array (node) location, and because rotating persons must
c  be added to UFT and FAIP/OTHER gains, which are already in 'node'.
c
      do 46 i=1,5
        node(i,duty,asdgrp(i),flyacc(i),1)=rote(i)+
     $     node(i,duty,asdgrp(i),flyacc(i),1)
46    continue
c
      go to 35
c  ************ end of loop that reads data **********
c*********************************************************************
40    continue
c
      close (5)
c
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c*********************************************************************
c*********************************************************************
cc                                                                  cc
cc   SECTION IV                                                     cc
cc                                                                  cc
cc   CALCULATIONS AND 'SANITY' CHECKS                               cc
cc                                                                  cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  These checks compare the quantity of individuals available in the
c  network with the requirements (side constraints) for each time
c  period.
c  If availability is not within the upper and lower requirements
c  bounds, execution is terminated.
c**********************************************************************
      open(unit=9,file='sanity.out',status='new')
c*********************************************************
c  OUTPUT SOME PARAMETERS TO SANITY.OUT
C
      WRITE (9,1000)
1000  FORMAT(//' INPUT PARAMETERS'/)
      write(9,1001),dutdur(1),dutdur(2),dutdur(3),dutdur(4)
1001  format(' DUTY TYPE:',10X,'FLY',4X,'SUP',4X,'AFIT',3X,
     $ 'PME'/' ','duty duration:'6X,I3,4X,I3,4X,I3,3X,I3)
      WRITE(9,1002)
1002  FORMAT(//' GATE COSTS:   GATE TIME    ASD    MISS COST'/)
      DO 2000 I=1,GOALS
```

```
        WRITE(9,1003)FLY(I),ASD(I),COST(I)
1003    FORMAT(' ',14X,I5,8X,I3,4X,I4)
2000  CONTINUE
      WRITE(9,1004)
1004  FORMAT(//' ATTRITION ADJUSTMENTS'/
     $ ' YEAR  DUTY  ASD   FLY    ADJUSTMENT'/
     $ ' ----  ----  ---   ---    ----------')
      DO 2001 TIME=1,5
        DO 2001 DUTY=1,2
          DO 2001 ASDYG=6,18
            DO 2001 FLYcre=6,11
              IF (CHANGE(TIME,DUTY,ASDYG,FLYcre).NE.0)
     $  WRITE(9,1005)TIME,DUTY,ASDYG,FLYcre,
     $   CHANGE(TIME,DUTY,ASDYG,FLYcre)
1005  FORMAT(' ',I4,2X,I4,2X,I3,3X,I3,4X,F5.0)
2001  CONTINUE
C
C**************************************************************
c  Calculate time-on-station totals.
c  i = time period in the past (half-year increments)
c
      do 88 duty=1,4
        do 88 i=1,9
          sumdas(duty) = sumdas(duty) + totdas(i,duty)
          sumtos(i) = sumtos(i) + totdas(i,duty)
88    continue
c
c  write date arrived station info to sanity.out file
c
      write(9,880) fy(1)-4,fy(1)-4,fy(1)-3,fy(1)-2,fy(1)-1
880   format(//' ARRIVED-STATION STATISTICS'/' (number that arrived',
     $  ' at initial duty station'/' during indicated time periods.'/
     $  ' 1st column = fy 1st half, 2nd column = fy 2nd half)'//
     $  '  duty  before'/
     $  '  type  ',1x,'FY',I2,2x,4(1x,'FY',I2,5x),'total'/
     $  '  ----  ','------',1x,4('---------',1x),'-----')
      do 77 j=1,4
        write(9,881) j,totdas(9,j),totdas(8,j),totdas(7,j),
     $    totdas(6,j),totdas(5,j),totdas(4,j),totdas(3,j),
     $    totdas(2,j),totdas(1,j),sumdas(j)
77    continue
881   format (' ',4x,I1,3x,F4.0,2x,4(F4.0,1x,F4.0,1x),1x,F5.0)
c
      write(9,886),sumtos(9),sumtos(8),sumtos(7),sumtos(6),sumtos(5),
     $  sumtos(4),sumtos(3),sumtos(2),sumtos(1)
886   format (' ','total',2x,I4,3x,4(I3,2x,I3,2x))
c
c*******************************************************************
c  sum rotations by year and INITIAL asd group (rotbya), by
c  INITIAL asd group (totasd), by initial duty type (gtdut),
c  and by duty type and initial asd group (asdpct)
c
```

```
c  Note:  asd groups are those assigned by the model, not necessarily
c  the actual asd group (i.e. max modeled asd group is 18).
c
      do 51 asdYG=0,18
        do 51 duty=1,4
          do 51 time=1,5
            rotbya(time,asdyg)=rotbya(time,asdyg) +
     $        totrot(time,duty,asdyg)
            totasd(asdyg)=totasd(asdyg) +
     $        totrot(time,duty,asdyg)
            gtdut(duty)=gtdut(duty) +
     $        totrot(time,duty,asdyg)
            asdsum(duty,asdyg)=asdsum(duty,asdyg)+
     $        totrot(time,duty,asdyg)
51    continue
c
c  calculate 'asdpct' (makeup of initial duty types, by asd group)
c
      do 52 asdyg=0,18
        do 52 duty=1,4
          if (gtdut(duty).ne.0.) then
            asdpct(duty,asdyg)=asdsum(duty,asdyg)/
     $        gtdut(duty)*100.
          endif
52    continue
c
c  write 'asdpct' statistics to sanity.out
c
      write(9,882)
882   format('1'//,' DUTY COMPOSITION STATISTICS'/' NOTE:  these '
     $  'stats represent the situation after the model'/' has',
     $  ' forced certain duty, ASD, and gate credit combinations'
     $  //' (columns sum ',
     $  'to 100 percent)'//' ','              percent of total ',
     $  'personnel in each duty'/' ',14x,'type that belong to ',
     $  'each asd group'//
     $  ' ','asd group       fly        sup        afit       pme'/
     $  ' ','---------       ---        ---        ----       ---')
      do 74 asdyg=0,18
        write(9,883) asdyg,asdpct(1,asdyg),asdpct(2,asdyg),
     $    asdpct(3,asdyg),asdpct(4,asdyg)
74    continue
c*****************************************************************
c  calculate 'dutpct' (distribution of each initial asd group, by
c  duty).
c
      do 50 duty=1,4
        do 50 asdyg=0,18
          if (totasd(asdyg).ne.0) then
            dutpct(duty,asdyg)=asdsum(duty,asdyg)/
     $        totasd(asdyg)*100.
          endif
```

```
50    continue
c
      write(9,885)
885   format(///' (rows sum to 100 percent)'//
     $  ' ',14x,'percent of total personnel in each'/
     $  ' ',14x,'asd group that are in each dutytype'//
     $  ' ','asd group',7x,'fly',7x,'sup',7x,'afit',6x,'pme'/
     $  ' ',8('-'),8x,3('-'),7x,3('-'),7x,4('-'),6x,3('-'))
      do 89 i=0,18
        write(9,883) i,dutpct(1,i),dutpct(2,i),dutpct(3,i),
     $    dutpct(4,i)
89    continue
883   format(' ',3x,I2,10x,4(F4.0,6x))
c*********************************************************************
c  calculate 'flypct', which is the makeup of each initial asd year
c  group broken out by flying credit.
c
c  Note:  breakout of asd groups by flycre values uses the flycre
c  values as assigned by the model, not the actual values.
c
      do 73 asdyg=0,18
        do 73 flycre=0,11
          if (totasd(asdyg).ne.0) then
            flypct(asdyg,flycre)=flytot(asdyg,flycre)/
     $        totasd(asdyg)*100.
          endif
73    continue
c
c  write 'flypct' statistics to sanity.out
c
      write(9,888)
888   format(///' ','ASD GROUP COMPOSITION STATISTICS'/' NOTE:  ',
     $  'These numbers are also after the model'/' has forced ',
     $  'certain combinations'/' (rows sum ',
     $  'to 100 percent)'//' ',12x,'% of each asd group by gate ',
     $  'credit accumulated'//
     $  ' asd group   6 yrs  7 yrs  8 yrs  9 yrs  10 yrs 11 yrs'/
     $  ' ---------   -----  -----  -----  -----  ------ ------')
      do 75 asdyg=0,18
      write(9,884) asdyg,flypct(asdyg,6),flypct(asdyg,
     $  7),flypct(asdyg,8),flypct(asdyg,9),flypct(asdyg,
     $  10),flypct(asdyg,11)
75    continue
884   format(' ',4x,I2,6x,6(F4.0,3x))
c
c  calculate rotations-to-date
c
      do 53 time=1,5
        do 53 i=1,time
          rotetd(time)=rotetd(time)+gtrote(i)
53    continue
c*********************************************************************
```

```
c  this is just another 'angle' on the same numbers:
c  'tinput(time)' accumulates the total rotations into the network,
c  including 'gains', for a particular time.
c  'totdut(time,duty)' accumulates the same information, but
c  broken out by duty type individuals are rotating from.
c
      do 54 time=1,5
        tinput(time)=gains(time)
        do 54 duty=1,4
          do 54 asdyg=0,18
            tinput(time)=tinput(time) +
     $        totrot(time,duty,asdyg)
            totdut(time,duty)=totdut(time,duty)+
     $        totrot(time,duty,asdyg)
54    continue
c******************************************************************
c  calculate losses due to attrition for each asd group and year.
c  Attrition of UFT gains is assumed to be
c  negligible.  Attrition of FAIP/other gains is calculated
c  separately (below).  Also, attrition of the initial (year 1)
c  18 year asd group is done separately (below).
c
c  cumulative attrition for each year based on current
c  (attained) asd group
c
      do 55 time=1,5
        do 55 asdyg=0,18
          oldasd=asdyg+1-time
          if (oldasd.lt.0) oldasd=0
          catrit(time,asdyg)=ANINT(totasd(oldasd)*
     $      (1-ccrasd(time,asdyg)))
55    continue
c
c  yearly attrition values (non-cumulative).  These are calculated
c  'round-about' by subtracting cumulative attrition values.
c
      do 57 time=1,5
        do 57 asdyg=1,18
          atrit(time,asdyg)=catrit(time,asdyg) -
     $      catrit(time-1,asdyg-1)
57    continue
c*****************************************************************
c  asd group 18 attrition must also include groups that entered
c  the 18+ group in past years.  For year 3, this includes
c  individuals from the original asd group 17; for year 4, this
c  includes original asd groups 16 & 17; for year 5, this
c  includes original asd groups 15, 16, & 17.
c
      do 170 time=3,5
c          atrit(time,18)=ANINT(atrit(time,18)-catrit(time-1,18))
c          catrit(time,18)=ANINT(catrit(time,18)-
c     $      catrit(time-1,18))
```

```
      do 170 oldasd-18-time+2,17
        atrit(time,18)=ANINT(atrit(time,18)+totasd(oldasd)*
     $      (1 ccrasd(time,18)))
        catrit(time,18)=ANINT(catrit(time,18)+totasd(oldasd)*
     $      (1-ccrasd(time,18)))
170   continue
c*****************************************************************
c  asd group 18 attrition must also include attrition of those who
c  started in year group 18 (and above).  Due to the method used
c  to calculate attritions, these individuals have been 'ignored'
c  thus far.
c
      do 58 time=1,5
        catr18(time)=ANINT(totasd(18)*(1-ccr18(time)))
58    continue
c
      do 59 time=1,5
        aatr18(time)=catr18(time)-catr18(time-1)
59    continue
c******************************************************************
c  calculate attrition for faip gains
c
      do 60 time=1,5
        ccrfa(time,4)=1.0
60    continue
c
      do 61 time=2,5
        do 61 newasd=5,4+time-1
          ccrfa(time,newasd)=1.0
          do 61 oldasd=5,newasd
            i=newasd-oldasd
            ccrfa(time,newasd)=ccrfa(time,newasd) *
     $        conasd(time-i,oldasd)
61    continue
c
      do 62 time=2,5
        do 62 asdyg=5,4+time-1
          catrfa(time,asdyg)=ANINT((1-ccrfa(time,asdyg))
     $      * faip(time+4-asdyg))
62    continue
c
      do 64 time=2,5
        do 64 asdyg=5,4+time-1
          atrfa(time,asdyg)= catrfa(time,asdyg)-
     $     catrfa(time-1,asdyg-1)
64    continue
c*****************************************************************
c  total attrition for each time period and asd group
c
      do 65 time=1,5
        do 65 asdyg=1,18
          atrflo(time,asdyg)=atrit(time,asdyg)
```

```
65    continue
c
      do 66 time=1,5
        atrflo(time,18)=atrflo(time,18)+aatr18(time)
66    continue
c
      do 67 time=2,5
        do 67 asdyg=5,4+time-1
          atrflo(time,asdyg)=atrflo(time,asdyg)+
     $      atrfa(time,asdyg)
67    continue
c***********************************************************************
c  Calculate attrition values by duty,asd group,flycre combos.
c  These are the negative external flows assigned to the appropriate
c  network nodes.
c  MODEL ASSUMPTION:  Attrition is negligible for asd groups LT 6.
c
      do 174 time=1,5
        do 174 duty=1,2
          do 174 asdyg=6,18
            if (asdyg.gt.11) then
              maxfly=11
            else
              maxfly=asdyg
            endif
            do 174 flycre=6,maxfly
              makeup=(dutmak(duty,asdyg)/100.) *
     $          (flymak(asdyg,flycre)/100.)
c
c  prevent attrition from being demanded from 'impossible'
c  fly nodes (flycre values lower than is possible)
c
              if (flycre.le.dutdur(1)+6) then
                if (duty.eq.1) makeup=0.
                if (duty.eq.2) makeup=flymak(asdyg,
     $            flycre)/100.
              endif
c
c  prevent attrition from being demanded from 'impossible'
c  nonfly nodes (flycre values greater than is possible)
c
              if (flycre.gt.asdyg-dutdur(2)) then
                if (duty.eq.1) makeup=flymak(asdyg,
     $            flycre)/100.
                if (duty.eq.2) makeup=0.
              endif
c
              atr(time,duty,asdyg,flycre)= ANINT(
     $          makeup*atrflo(time,asdyg)) +
     $          CHANGE(TIME,DUTY,ASDyg,FLYcre)
              totout=totout+atr(time,duty,asdyg,flycre)
              totatr(time)=totatr(time)+atr(time,duty,
```

```
     $          asdyg,flycre)
174   continue
c
c  attrition to date
c
      do 69 time=1,5
        do 69 i=1,time
          atrtd(time)=atrtd(time)+totatr(i)
69    continue
c*********************************************************************
c
c
c
c  Sum all personnel that are projected to be in the system for
c  each time period (rotations to date + gains to date -
c  cumulative attrition to date
c
      do 70 time=1,5
          totflo(time)=rotetd(time)+gaintd(time)-atrtd(time)
70    continue
c
c*********************************************************************
c  Calculate minimum undermanned requirement (reqund) and
c  maximum overmanned requirement (reqove) for each duty type and
c  time period, based on input requirements (rqmt) and tolerances
c  (pctund and pctove).
c
      do 71 time=1,5
        do 71 duty=1,4
          reqove(time,duty)=ANINT(rqmt(time,duty)*
     $      (1.0+pctove(time,duty)/100.0))
          reqund(time,duty)=ANINT(rqmt(time,duty)*
     $      (1.0-pctund(time,duty)/100.0))
71    continue
c
c  calculate the number of individuals that are already in the MWS
c  but their first rotation 'into the network' is scheduled
c  downstream.  These individuals are
c  filling manning positions, but the network doesn't know about
c  them 'yet'.  Values of 'adjust(time)' are used to adjust
c  side constraint manning requirements.
c
      do 72 time=1,5
        do 72 duty=1,4
          do 72 i=time+1,5
            do 72 asdyg=0,18
              adjust(time,duty)=adjust(time,duty)+
     $          totrot(i,duty,asdyg)
72    continue
c
c  Adjusted min/max manning level is undermanned/overmanned
c  requirement minus downstream rotations (people filling
```

```
c  positions but not yet in the network model).
c
      do 171 time=1,5
        do 171 duty=1,4
          adjund(time,duty)=reqund(time,duty)-
     $      adjust(time,duty)
          adjove(time,duty)=reqove(time,duty)-
     $      adjust(time,duty)
171     continue
c
c  calculate total under-/over-manned requirements for each
c  time period.
c
c  Sum adjusted requirements across duty types to obtain total
c  undermanned/overmanned requirements for each time period
c  (tadjun/tadjov).
c
      do 172 time=1,5
        do 172 duty=1,4
          tadjov(time)=tadjov(time)+adjove(time,duty)
          tadjun(time)=tadjun(time)+adjund(time,duty)
172   continue
c
c  Numbers for sanity file output
c
      do 173 time=1,5
        do 173 duty=1,4
          totund(time)=totund(time)+reqund(time,duty)
          totove(time)=totove(time)+reqove(time,duty)
          totadj(time)=totadj(time)+adjust(time,duty)
173   continue
c*******************************************************************
c  output numbers to 'sanity.sal' data file for verification checks.
c
c************************************************************
c  totrot values
c
      write (9,900) fy(1),fy(2),fy(3),fy(4),fy(5)
900   format ('1','YEAR:',10x,5(I4,7x)/17x,5('-------',4x))
c
      write (9,970)
970   format (' ','TOTAL ROTATIONS'/' ','grouped by 'from' ',
     $  'duty and 'initial' asd group')
c
c  i=from duty, j=initial asd group
c
      do 79 i=1,4
        write(9,856)
        do 79 j=0,18
          write(9,908) i,j,totrot(1,i,j),totrot(2,i,j),
     $      totrot(3,i,j),totrot(4,i,j),totrot(5,i,j)
79    continue
```

```
856   format (' ',' ')
908   format (' ','DUTY',I2,'; ASD',I3,2x,5(F6.0,5x))
C***********************************************************
C  ROTATIONS THAT HAVE ALREADY FAILED TO MEET THEIR GATES
C
      WRITE(9,940) FY(1),FY(2),FY(3),FY(4),FY(5)
940   FORMAT (//' ','ROTATIONS THAT HAVE MISSED THEIR GATES'
     $  //' YEAR:',10X,5(I4,7X)/' ',17X,5('-------',4X)/)
      WRITE(9,902) '1ST GATE',FAILG1(1),FAILG1(2),FAILG1(3),
     $  FAILG1(4),FAILG1(5)
      WRITE(9,902) '2ND GATE',FAILG2(1),FAILG2(2),FAILG2(3),
     $  FAILG2(4),FAILG2(5)
      WRITE(9,902) '3RD GATE',FAILG3(1),FAILG3(2),FAILG3(3),
     $  FAILG3(4),FAILG3(5)
902   FORMAT (' ',A8,10X,5(I4,7X))
C
C***********************************************************
c  available personnel
c
      write (9,900) fy(1),fy(2),fy(3),fy(4),fy(5)
c
      write (9,903)
903   format (' ','AVAILABLES'/' ',10('-')//' ','GAINS'/)
c
      write (9,905) uft(1),uft(2),uft(3),uft(4),uft(5)
905   format (' ','UFT GAINS:',5x,5(I5,6x))
c
      write (9,906) faip(1),faip(2),faip(3),faip(4),faip(5)
906   format (' ','FAIP GAINS:',4x,5(I5,6x))
c
      write (9,907) gains(1),gains(2),gains(3),gains(4),gains(5)
907   format (' ','TOTAL GAINS:',3x,5(F6.0,5x))
c
      write (9,935) gaintd(1),gaintd(2),gaintd(3),
     $  gaintd(4),gaintd(5)
935   format (' ','GAINS TO DATE: ',5(F6.0,5x)/)
c
      write (9,912)
912   format (' ','ROTATIONS'/)
c
      do 80 duty=1,4
        write(9,910) duty,totdut(1,duty),
     $     totdut(2,duty),totdut(3,duty),
     $     totdut(4,duty),totdut(5,duty)
80    continue
910   format (' ','DUTY    ',I2,':',4x,5(F6.0,5x))
c
      write(9,934) gtrote(1),gtrote(2),gtrote(3),
     $  gtrote(4),gtrote(5)
934   format (' ','TOTAL ROTES:',3x,5(F6.0,5x))
c
      write(9,936) rotetd(1),rotetd(2),rotetd(3),
```

```
     $  rotetd(4),rotetd(5)
936   format (' ','ROTES TO DATE:',5(F7.0,4x)//)
c
      write(9,913) totatr(1),totatr(2),totatr(3),
     $  totatr(4),totatr(5)
913   format (' ','ATTRITION:',5x,5(F6.0,5x))
c
      write(9,939) atrtd(1),atrtd(2),atrtd(3),
     $  atrtd(4),atrtd(5)
939   format (' ','ATRIT TO DATE: ',5(F6.0,5x)//)
c
      write(9,915) totflo(1),totflo(2),totflo(3),totflo(4),
     $  totflo(5)
915   format (' ','TOT AVAILABLE: ',5(F6.0,5x)/)
c
      write(9,916)
916   format (' ','tot available = gains to date + rotations ',
     $  'to date - attrition to date'///)
c*******************************************************************
c  requirements
c
      write(9,901) fy(1),fy(2),fy(3),fy(4),fy(5)
901   format (///' YEAR:',10x,5(I4,7x)/17x,5('-------',4x))
c
      write(9,917)
917   format (' ','REQUIREMENTS'/' ',12('-')//' ',
     $  'BASELINE REQUIREMENTS (from R.M.D.)'/)
c
      do 81 duty=1,4
        write(9,910) duty,rqmt(1,duty),rqmt(2,duty),
     $    rqmt(3,duty),rqmt(4,duty),rqmt(5,duty)
81    continue
c
      write(9,919) totreq(1),totreq(2),totreq(3),totreq(4),
     $    totreq(5)
919   format (' ','TOTAL RQMTS:',3x,5(F6.0,5x))
c
      write(9,920)
920   format (' '/' ','UNDERMANNED (MIN) REQUIREMENTS'/)
c
      do 82 duty=1,4
        write(9,910) duty,reqund(1,duty),
     $    reqund(2,duty),reqund(3,duty),
     $    reqund(4,duty),reqund(5,duty)
82    continue
c
      write (9,931) totund(1),totund(2),totund(3),
     $   totund(4),totund(5)
931   format (' ','TOTAL UNDER:',3x,5(F6.0,5x))
c
      write (9,921)
921   format (' '/' ','OVERMANNED (MAX) REQUIREMENTS'/)
```

```
      do 83 duty=1,4
        write(9,910) duty,reqove(1,duty),
     $     reqove(2,duty),reqove(3,duty),
     $     reqove(4,duty),reqove(5,duty)
83    continue
c
      write(9,932) totove(1),totove(2),totove(3),totove(4),
     $  totove(5)
932   format (' ','TOTAL OVER:',4x,5(F6.0,5x))
c
      write(9,918)
918   format (' '/' ','ADJUSTMENT (FUTURE ROTATIONS)'/)
c
      do 84 duty=1,4
        write (9,910) duty,adjust(1,duty),
     $     adjust(2,duty),adjust(3,duty),
     $     adjust(4,duty),adjust(5,duty)
84    continue
c
      write(9,933) totadj(1),totadj(2),totadj(3),
     $  totadj(4), totadj(5)
933   format (' ','TOTAL ADJUST:',2x,5(F6.0,5x))
c
      write(9,890)
890   format(' '/' ADJUSTED UNDERMANNED REQUIREMENTS'/
     $  ' (undermanned rqmts reduced for downstream rotations)'/)
c
      do 177 duty=1,4
        write(9,910) duty,adjund(1,duty),adjund
     $     (2,duty),adjund(3,duty),adjund(4,duty),
     $     adjund(5,duty)
177   continue
c
      write(9,891)
891   format (' '/' ADJUSTED OVERMANNED REQUIREMENTS'/
     $  ' (overmanned rqmts reduced for downstream rotations)'/)
c
      do 178 duty=1,4
        write(9,910) duty,adjove(1,duty),adjove(2,
     $     duty),adjove(3,duty),adjove(4,duty),
     $     adjove(5,duty)
178   continue
c
      write(9,922)
922   format (' '/' ','ADJUSTED TOTAL REQUIREMENTS'/
     $  ' ','(Requirements reduced for downstream rotations)'/)
c
      write(9,923) tadjun(1),tadjun(2),tadjun(3),
     $  tadjun(4),tadjun(5)
923   format (' ','UNDERMANNED:',3x,5(F6.0,5x))
c
      write(9,876) totreq(1)-totadj(1),totreq(2)-totadj(2),
```

```
     $  totreq(3)-totadj(3),totreq(4)-totadj(4),
     $  totreq(5)-totadj(5)
876   format (' ','BASELINE:',6x,5(F6.0,5x))
c
      write(9,924) tadjov(1),tadjov(2),tadjov(3),
     $  tadjov(4),tadjov(5)
924   format (' ','OVERMANNED:',4x,5(F6.0,5x))
c
      write(9,925)
925   format (' '/' ','tot available corresponds to the',
     $  ' flows in the network'/
     $  '  for each year and should fall between undermanned '/
     $  '  and overmanned adjusted total requirements.')
c*****************************************************************
c  write attrition info to sanity.out
c
      write(9,830)
830   format ('1','ATTRITION'//
     $' asd    gate   year 1    year 2    year 3    year 4    year 5'/
     $' group  time   fly sup   fly sup   fly sup   fly sup   fly sup'/
     $' -----  ----   --- ---   --- ---   --- ---   --- ---   --- ---')
      do 175 asdyg=6,18
        if (asdyg.gt.11) then
          maxfly=11
        else
          maxfly=asdyg
        endif
        do 175 flycre=6,maxfly
            write(9,831) asdyg,flycre,atr(1,1,asdyg,
     $  flycre),atr(1,2,asdyg,flycre),atr(2,1,asdyg,
     $  flycre),atr(2,2,asdyg,flycre),atr(3,1,asdyg,
     $  flycre),atr(3,2,asdyg,flycre),atr(4,1,asdyg,
     $  flycre),atr(4,2,asdyg,flycre),atr(5,1,asdyg,
     $  flycre),atr(5,2,asdyg,flycre)
831   format (' ',I5,I6,3x,5(F4.0,F4.0,1x))
175   continue
      close(9)
c*********************************************************************
c  print info to screen
c
      print *,' '
      print *,'         year','   lower bound      ',
     $ 'available       ','upper bound'
      print *,'         ----','   -----------      ',
     $ '---------       ','-----------'
c
      do 86 time=1,5
        print *,time,tadjun(time),totflo(time),tadjov(time)
86    continue
c
      do 87 time=1,5
        if (tadjun(time).GT.totflo(time)) then
```

```
          print *,' '
          print *,'insufficient personnel available in year ',time
          print *,'make one of the following adjustments:'
          print *,'  decrease manning requirements'
          print *,'  increase undermanning tolerances'
          print *,'  increase UFT, FAIP/other gains'
          print *,'  decrease attrition rates'
          flag=1
        endif
        if (tadjov(time).LT.totflo(time)) then
          print *,' '
          print *,'excessive personnel available in year ',time
          print *,'make one of the following adjustments:'
          print *,'  increase manning requirements'
          print *,'  increase overmanning tolerances'
          print *,'  decrease UFT, FAIP/other gains'
          print *,'  increase attrition rates'
          flag=1
        endif
87    continue
c**********************************************************************
c  abort execution if sanity checks are not passed
c
      if (flag.EQ.1) then
        print *,' '
        print *,'execution aborted'
        stop
      else
        print *,' '
        print *,'sanity checks passed....building datafiles'
      endif
c
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c**********************************************************************
c**********************************************************************
cc                                                                  cc
cc   SECTION V                                                      cc
cc                                                                  cc
cc   ASSIGN NODE NUMBERS AND EXTERNAL FLOWS;                        cc
cc   WRITE TO NETSID DATA FILE 1                                    cc
cc                                                                  cc
c**********************************************************************
c
c  loop to assign successive node numbers to all modeled combinations
c  of time, duty, asd group, and flycre and write node numbers/
c  external flows to data file 1.
c
      open (unit=6,file='FOR001.dat',status='new')
c
      nodnum=0
c
      do 90 time=1,5
```

```
         do 90 duty=1,4
           do 90 asdyg=0,18
c
             if (asdyg.LT.6) then
               minfly=asdyg
             else
               minfly=6
             endif
c
             if (asdyg.GT.11) then
               maxfly=11
             else
               maxfly=asdyg
             endif
c
             do 90 flycre=minfly,maxfly
               extflo=0
               flag1=0
c
c  asd groups LT 6 years can only be in flying jobs (model constraint)
c
               if (duty.NE.1.AND.asdyg.LT.6) flag1=1
c
c  asd groups GT 13 years can not be in AFIT (model constraint)
c
               if (duty.EQ.3.AND.asdyg.GT.13) flag1=1
c
c  PME/ASTRA is only available to asd groups 7-8 (astra),
c  13-15 (iss), and 18 + (sss). These are end-of-tour ASD
c  values. (model constraint)
c
               if (duty.eq.4) then
                 if (asdyg.LT.7) flag1=1
                 if (asdyg.GT.8.AND.asdyg.LT.13) flag1=1
                 if (asdyg.GT.15.AND.asdyg.LT.18) flag1=1
               endif
c
               if (flag1.NE.1) then
                 nodnum=nodnum+1
                 node(time,duty,asdyg,flycre,0)=nodnum
                 extflo=node(time,duty,asdyg,flycre,1)
     $             - ATR(TIME,DUTY,ASDyg,FLYcre)
               endif
c
               if (extflo.NE.0.) then
                 write (6,400) nodnum,extflo
               endif
400            format (' ',I5,F10.2)
c
90     continue
c
       maxnod=nodnum
```

```
      print *,'total nodes - ',maxnod+5
c*******************************************************************
c  write the negative external flow at the end of the network
c  'sink' node to data file 1.  External flow at this node equals
c  -(inflows - outflows), where inflows = 'totin' = gains +
c  rotations into the network, and outflows = 'totout' = attrition.
c
      totin=rotetd(5)+gaintd(5)
      print *,' total flows into network = ',totin
      print *,' total attrition out of network = ',totout
      write (6,400) maxnod+5, -(totin-totout)
      print *,' end-of-network sink = ',-(totin-totout)
c
c  Note:  the 5 end-of-network nodes are not stored in array 'node'
c*******************************************************************
      close (6)
c
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c*********************************************************************
c*********************************************************************
cc                                                                  cc
cc   SECTION VI                                                     cc
cc                                                                  cc
cc   ASSIGN ARC NUMBERS, ARC COSTS, AND ARC BOUNDS;                 cc
cc   WRITE TO DATA FILE 2;                                          cc
cc   WRITE BEGIN NODE, END NODE, AND ARC NUMBERS TO NODEARC.OUT FILE cc
cc                                                                  cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  NOTE:  the rows in FILE 2 must be ordered by arc number.
c
c*********************************************************************
c
c  loop to calculate combinations of beginning and ending nodes and
c  assign corresponding arc numbers
c
      open (unit=7,file='FOR002.dat',status='new')
      open (unit=8,file='nodearc.out',status='new')
c
      write (8,500)
500   format (' ','  begin',17x,'  end',19x,'   arc')
      write (8,510)
510   format (' ',' node time duty asd fly',
     $  '  node time duty asd fly',3x,'num cost bound'/)
c
      arcnum=0
c******************************************************************
      do 110 enddut=1,4
c
c  Record max arc number for each end duty type.
c
        maxarc(enddut-1)=arcnum
```

```
c
       do 110 time=1,5
         do 110 duty=1,4
c
c  AFIT can only rotate to SUP
c
           if (duty.eq.3.and.enddut.ne.2) goto 111
c
c  PME cannot rotate to PME
c
           if (duty.eq.4.and.enddut.eq.4) goto 111
c
           do 110 asdyg=0,18
C
c  DO NOT ALLOW ASD GROUPS .LT. 6 TO ROTATE INTO NONFLY DUTIES.
C  THIS IS ALREADY ACCOMPLISHED FOR YEARS 1-5 BY STRUCTURAL
C  CONSTRAINTS:  THE CORRESPONDING NODES SIMPLY DO NOT EXIST.
C  HOWEVER, THIS IS NOT THE CASE FOR THE END-OF-NETWORK
C  NODES/ARCS.
C
             IF (ENDDUT.NE.1.AND.ASDyg.LT.6) GOTO 111
c
             if (asdyg.LT.6) then
               minfly=asdyg
             else
               minfly=6
             endif
c
             if (asdyg.GT.11) then
               maxfly=11
             else
               maxfly=asdyg
             endif
c
c  All duty assignments for ASD group 18 are given a duration of
c  1 year.  This was necessary because of the large number of
c  attritions out of this ASD year group (due, in part, to 20-
c  year retirements).
c
             dur = dutdur(enddut)
             if (asdyg.eq.18) dur = 1
c
             do 110 flycre=minfly,maxfly
c
c  Any node assigned a nonzero node number (in previous do-loop)
c  is examined as a possible beginning node.
c  For those that lead to possible (nonzero) end nodes
c  via assignment arcs, arc numbers and corresponding end nodes are
c  calculated and written to data file 2.
c
               begnod=node(time,duty,asdyg,flycre,0)
c
```

```
c  must insure only nonzero beginning nodes are assigned arcs
c
              if (begnod.ne.0) then
c
c  calculate asd group at end of assignment
c
                if (asdyg+dur.GT.18) then
                  NEWasd=18
                else
                  NEWasd=asdyg+dur
                endif
c
c  calculate flycre at end of assigment
c
                if (enddut.EQ.1) then
                  if (flycre+dur.GT.11) then
                    NEWfly=11
                  else
                    NEWfly=flycre+dur
                  endif
                else
                  NEWfly=flycre
                endif
c
c  any assignments that would terminate beyond the time horizon
c  of the model are routed to node maxnod + end duty
c
                endtim=time+dur
                if (endtim.GT.5) then
                  if(enddut.eq.3.and.newasd.gt.13) goto 111
                  if(enddut.eq.4) then
                    if(newasd.lt.7) goto 111
                    if(newasd.gt.8.and.newasd.lt.13) goto 111
                    if(newasd.gt.15.and.newasd.lt.18) goto 111
                  endif
                  endnod=maxnod+enddut
                else
                  endnod=node(endtim,enddut,NEWasd,NEWfly,0)
                endif
c
c  arc numbers are assigned sequentially to begin node/end node
c  combinations where the end node is nonzero--that is, it 'exists'
c  in the network model and not just in the node array.
c  Record arc begin and end node information in array 'arc'.  Note
c  that array arc holds projected end-tour information even if
c  the assignment theoretically ends beyond the time horizon of the
c  network.
c
                if (endnod.NE.0) then
                  arcnum=arcnum+1
                  arc(arcnum,1)=begnod
                  arc(arcnum,2)=endnod
```

```
                    arc(arcnum,5)=time
                    arc(arcnum,6)=endtim
                    arc(arcnum,7)=asdyg
                    arc(arcnum,8)=newasd
                    ARC(ARCNUM,9)=FLYcre
                    ARC(ARCNUM,10)=NEWFLY
c
c  assign arc costs
c
c  arc costs need to be ordered high-to-low, since the first time
c  the if-then is satisfied, the associated cost is assigned and
c  no other rqmts/policies/goals are examined.  (HIGH-TO-LOW
C  ORDERING IS NOT NECESSARY IF ONLY ONE OF THE COSTS APPLIES
C  TO EACH ARC.)  This way, the
c  highest cost applicable is assigned to each arc.  However,
c  zero-costs can be interspersed with positive cost values.
c  Any assignments made during the 5-year time horizon that
c  result in non-attainment of gate requirements or goals by the
c  end of that assignment (even though the assignment ends beyond
c  the model time horizon) will contribute to objective function
c  costs.  However, it is conceivable that some non-flying
c  assignments (with zero cost assigned by the model) ending
c  beyond the time horizon could cause a person to reach an
c  asd group/flycre position from which he/she eventually
c  cannot attain a gate requirement or goal, even if subsequently
c  assigned to a flying duty.  Incremental gate goals, with
c  associated costs, can help insure downstream non-attainment
c  of requirements and goals are reflected in objective function
c  costs (and, therefore, optimal solution).
c
                    arcost=0
c
                    j=1
112   continue
                    if (cost(j).eq.0) then
                      IF (J.LT.GOALS) THEN
                        j=j+1
                        goto 112
                      ELSE
                        GOTO 114
                      ENDIF
                    endif
c
                    if (NEWasd.GE.asd(j).AND.NEWfly.LT.
     x               fly(j)) then
                      arcost = cost(j)
                      goto 114
                    else
                      if (j.LT.goals) then
                        j=j+1
                        goto 112
                      endif
```

```
                    endif
114   continue
                    arc(arcnum,3)=arcost
c
c  assign arc bounds
c  **** arcs are all unbounded ****
c
                    arc(arcnum,4)=-1
c
c  convert bound and cost values to real numbers for output to
c  NETSID data file.
c
                    rbound=arc(arcnum,4)
                    rcost=arc(arcnum,3)
c
                    write (7,571) arcnum,begnod,endnod,
     $               rbound,rcost
571                 format (' ',I5,2I6,2F10.2)
c
c
                    write (8,560) begnod,time,duty,asdyg,
     $              flycre,endnod,endtim,enddut,newasd,newfly,
     $              arcnum,arcost,rbound
560                 format (' ',2(I6,I5,I5,I4,I4),I6,I5,2x,F5.1)
c
                  endif
                endif
111     continue
c
110    continue
c**********************************************************************
c  Record max arc number for last end duty
c
      maxarc(enddut-1)=arcnum
c**********************************************************************
c  Arcs to End-of-network Sink node
c
      arcnum=maxarc(4)
      do 118 i=1,4
        arcnum=arcnum+1
        write(7,571) arcnum,maxnod+i,maxnod+5,-1.0,0.
        write(8,560) maxnod+i,98,i,0,0,maxnod+5,99,0,0,0,
     $    arcnum,0,-1.0
118   continue
c*****************************************************************
c
      do 117 i=1,4
        print *,'max arc number for enddut ',i,' is ',maxarc(i)
117   continue
c
      print *, 'total arcs = ',maxarc(4)+4
c*****************************************************************
```

```
      close (7)
      close (8)
c
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c*********************************************************************
c*********************************************************************
cc                                                                  cc
cc    SECTION VII                                                   cc
cc                                                                  cc
cc    ASSIGN CONSTRAINT NUMBERS TO MANNING REQUIREMENTS             cc
cc    AND EXPERIENCE LEVEL REQUIREMENTS;                            cc
cc    WRITE TO DATA FILE 4                                          cc
cc                                                                  cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  NOTE:  the rows in FILE 4 can be in any order.
c
c*********************************************************************
c
      open(unit=9,file='FOR004.dat',status='new')
c***************************************************************
C  MANNING LEVEL CONSTRAINTS (FYDP REQUIREMENTS ADJUSTED FOR
C  THE UNDER/OVER MANNING TOLERANCE
C
      constn=0
      do 130 time=1,5
        do 130 duty=1,4
          do 130 ctype=0,1
c
c  ctype=0 is a less-than constraint
c  ctype=1 is a greater-than constraint
c
            constn=constn+1
            constr(time,duty,ctype)=constn
            if (ctype.EQ.0) then
              ctypeC='L'
              rhs=adjove(time,duty)
            endif
            if (ctype.EQ.1) then
              ctypeC='G'
              rhs=adjund(time,duty)
            endif
            write (9,600) constn,rhs,ctypeC
600         format (' ',I5,F10.2,A1)
130    continue
c**************************************************************
c  Constraints for ASTRA,ISS,SSS
c
      do 132 time=1,5
        do 132 pmetyp=1,3
          do 132 ctype=0,1
            constn = constn + 1
```

```
            pmecon(time,pmetyp,ctype) = constn
            if (ctype.eq.0) then
              ctypec = 'L'
              if (pmetyp.eq.1) rhs = ANINT(ASTRA(time)*
     $          (1.+pctove(time,4)/100.))
              if (pmetyp.eq.2) rhs = ANINT(ISS(time)*
     $          (1.+pctove(time,4)/100.))
              if (pmetyp.eq.3) rhs = ANINT(SSS(time)*
     $          (1.+pctove(time,4)/100.))
            endif
c
            if (ctype.eq.1) then
              ctypec = 'G'
              if (pmetyp.eq.1) rhs = ANINT(ASTRA(time)*
     $          (1.-pctund(time,4)/100.))
              if (pmetyp.eq.2) rhs = ANINT(ISS(time)*
     $          (1.-pctund(time,4)/100.))
              if (pmetyp.eq.3) rhs = ANINT(SSS(time)*
     $          (1.-pctund(time,4)/100.))
            endif
      write (9,600) constn,rhs,ctypec
132   continue
c
c********************************************************************
c  Experience level constraints.
c
      do 131 time=1,5
        do 131 i=1,expreq
          constn=constn+1
          expcon(i,time)=constn
          if (exptyp(i).eq.0) then
            ctypec='L'
          else
            if (exptyp(i).eq.1) then
              ctypec='G'
            endif
          endif
          rhs = ANINT(exppct(i)/100.*(rqmt(time,expdut(i))-
     $      adjust(time,expdut(i))))
          write(9,600) constn,rhs,ctypec
131   continue
c
      close (9)
c
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c*********************************************************************
c*********************************************************************
cc                                                                  cc
cc    SECTION VIII                                                  cc
cc                                                                  cc
cc    DETERMINE WHICH CONSTRAINTS APPLY TO WHICH ARCS;              cc
cc    WRITE TO DATA FILE 3                                          cc
```

```
cc                                                                   cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  NOTE:  the rows in FILE 3 must be ordered by ARC NUMBER
c
c***********************************************************************
c
c  Note:  assumption is that all positive external flows are
c  received and assignments are made for a particular time
c  period before the corresponding constraints (manning requirements
c  and experience level rqmts) for that time period are applied.
c
      open (unit=8,file='FOR003.dat',status='new')
c
      do 150 enddut=1,4
c  step through arc numbers sequentially to insure file is ordered
c  by arc number
        do 150 arcnum=1+maxarc(enddut-1),maxarc(enddut)
          do 150 time=1,5
c
c  If begin time of arc <= current 'time' < end time of arc,
c  then constraints applicable to current year apply to the arc.
c
              if (arc(arcnum,5).LE.time.AND.
     $          arc(arcnum,6).GT.time) then
c
c  Manning level constraints.  All arcs except the four end-of-
c  network arcs have manning level constraints.
c
                write (8,700) constr(time,enddut,0),
     $            arcnum,1.0
                write (8,700) constr(time,enddut,1),
     $            arcnum,1.0
700             format (' ',I5,I6,F10.0)
c
c  ASTRA, ISS, and SSS constraints.  Note:  the network structure
c  is also constrained so that nodes for duty type 4 exist only
c  for the ranges of asd groups represented here.
c
                if (enddut.eq.4) then
                  if (arc(arcnum,8).ge.7.and.arc(arcnum,8).le.8)
     $             then
                    write(8,700) pmecon(time,1,0),arcnum,1.0
                    write(8,700) pmecon(time,1,1),arcnum,1.0
                  endif
                  if (arc(arcnum,8).ge.13.and.arc(arcnum,8).
     $             le.15) then
                    write(3,700) pmecon(time,2,0),arcnum,1.0
                    write(8,700) pmecon(time,2,1),arcnum,1.0
                  endif
                  if (arc(arcnum,8).ge.18) then
                    write(8,700) pmecon(time,3,0),arcnum,1.0
```

```
                    write(8,700) pmecon(time,3,1),arcnum,1.0
                  endif
                endif
c
c  Experience level constraints.  For 'less-than' constraints,
c  if arc begin asd .LE. 'expasd' then constraint applies.
c  For 'greater-than' constraints, if arc end asd. GE. 'expasd'
c  then constraint applies.
c
                do 151 i=1,expreq
                  if (exptyp(i).eq.0) then
                    if((expdut(i).eq.enddut).and.
     $              (arc(arcnum,7).le.expasd(i))) then
                      write(8,700) expcon(i,time),arcnum,1.0
                    endif
                  else
                    if(exptyp(i).eq.1) then
                      if((expdut(i).eq.enddut).and.
     $                (arc(arcnum,8).ge.expasd(i))) then
                        write(8,700) expcon(i,time),arcnum,1.0
                      endif
                    endif
                  endif
151             continue
c
              endif
150     continue
c
      close (8)
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c**********************************************************************
c**********************************************************************
cc                                                                   cc
cc   SECTION IX                                                      cc
cc                                                                   cc
cc   CALL NETSID TO COMPUTE OPTIMAL NETWORK SOLUTION                 cc
cc   NETSID PUTS OUTPUT IN FILES FOR007.DAT & FOR008.DAT             cc
cc                                                                   cc
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
      print *,'data files built...performing network optimization ',
     $  'routine'
      call NETSID
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c**********************************************************************
c**********************************************************************
cc                                                                   cc
cc   SECTION X                                                       cc
cc                                                                   cc
cc   CALL NETOUT TO CONVERT NETSID OUTPUT DATA TO                    cc
```

```
cc    'USABLE INFORMATION'                                        cc
cc                                                                cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
      print *,'optimization complete...building output tables'
      call Netout(maxarc,fy,node,atr,dutdur)
      print *,'processing complete'
      print *,'a list of nodes and arcs is in file NODEARC.OUT'
      print *,'optimal assignment tables are in file ROTEPLAN.OUT'
      print *,'miscellaneous optimization information is in ',
     $  'file FOR007.dat'
c
c
      end
c
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c*****************************************************************
c*****************************************************************
c  M. OLSON, thesis, DEC 87
c
c  subprogram to convert Netsid output data to USABLE INFORMATION
c  for the GATES program
c
c*****************************************************************
c
      subroutine NETOUT(maxarc,fy,node,atr,dutdur)
c
c*****************************************************************
c
c  VARIABLE DEFINITIONS AND DECLARATIONS
c
c  VARIABLES PASSED FROM 'GATES':
c
c  maxarc is dimensioned for the number of duty types and holds the
c   highest arcnumber representing an assignment to a particular
c   duty type
      integer maxarc(0:4)
c  fy is dimensioned for the number of time periods
      integer fy(1:5)
c  node is dimensioned to the full size of the network: time
c   periods, duty types, asd groups, flycre, and 'index', where
c   index = 0 = node number
c   index = 1 = node external flow (not including attrition)
      integer node(1:5,1:4,0:18,0:11,0:1)
c  atr is dimensioned for time periods, duty types, asd groups, and
c   flycre
      real atr(1:5,1:4,0:18,0:11)
c  dutdur is dimensioned for the number of duty types
      integer dutdur(1:4)
c
c
c  VARIABLES INTERNAL TO 'NETOUT':
```

```
c
c
      integer time, enddut, asdyg, flycre
      integer minfly, maxfly, index, arcnam
      integer nodnum, fnode, tnode, duty
      real flow, flocst, tgroup, gtotal, natrit
      real total, tatrit
c
c  the following hold rotation 'assignments' by time period
c  and end duty.  Each array represents a different level
c  of flying gate attainment:
c   gate0(time,enddut) = those who have not yet met any gates.
c   gate1y = have met the 1st gate
c   gate1n = failed to meet the 1st gate
c   gate2y = have met the 2nd gate
c   gate2n = failed to meet the 2nd gate (met gate 1)
c   gate3y = have met the 3rd gate
c   gate3n = failed to meet the 3rd gate (met gates 1 and 2)
      real gate0(1:5,1:4),gate1y(1:5,1:4),gate1n(1:5,1:4)
      real gate2y(1:5,1:4)
      real gate2n(1:5,1:4),gate3y(1:5,1:4),gate3n(1:5,1:4)
c  nassign holds the flow quantity and cost and is dimensioned for
c  from node (fnode), enddut, and index:
c   index = 1 = flow quantity
c   index = 2 = flow costs
      real nassign(1:5000,1:4,1:2)
c  dimension for number of duties:
      real tassign(1:4), nodenu(1:4)
      real assign(1:4), cost(1:4)
c  dimension for number of anticipated artificial flows and index.
c  artif holds 'solution' information on artificial flows:
c   index = 1 = fnode
c   index = 2 = tnode
c   index = 3 = flow
c   index = 4 = flocst
      real artif(1:100,1:4)
c  dimension for number of years +1 (for total), duty types +1
c  (attrition), asd groups, and flycre:
c  passign holds the proportion of assignments to each end duty
c   and attrition.
      real passign(1:6,1:5,0:18,0:11)
C
C  ROTEWC(TIME,ENDDUT,ASDYG,FLYCRE,INDEX) HOLDS ASSIGNMENTS THAT
C  HAVE A COST ASSOCIATED.
C   INDEX = 1 = COST ASSOCIATED
C   INDEX = 2 = # ASSOCIATED
C   INDEX = 3 = # FROM FLY DUTIES
C   INDEX = 4 = # FROM STAFF/SUP DUTIES
C   INDEX = 5 = # FROM AFIT
C   INDEX = 6 = # FROM PME
C
      REAL ROTEWC(1:5,1:4,0:18,0:11,1:6)
```

```
c
c*************************************************************************
c
c  CONVERT OUTPUT DATA TO 'INFORMATION';
c  WRITE TO FILE ROTEPLAN.OUT
c
      open (unit=5,file='for008.dat',status='old')
c
      k=0
c*************************************************************
c  ***** beginning of loop to read netsid output data *****
c
89    continue
      read (5,600,end=90) index,arcnam,fnode,tnode,flow,
     $  flocst
600   format (5X,I6,3(I8),2(E19.9))
c
c  artificial variable information is stored in artif(k,index)
c
      if (arcnam.eq.99999) then
        k=k+1
        artif(k,1)=fnode
        artif(k,2)=tnode
        artif(k,3)=flow
        artif(k,4)=flocst
        goto 89
      endif
c
c  network solution flows and costs are assigned to
c  nassign(fromnode,endduty,index), where index=1 holds the
c  flow quantities and index=2 holds the flow costs.
c
      do 95 enddut=1,4
        if (arcnam.GT.maxarc(enddut-1).AND.
     $    arcnam.LE.maxarc(enddut)) then
          nassign(fnode,enddut,1)=flow
          nassign(fnode,enddut,2)=flocst
        endif
95    continue
c
      go to 89
c
c  ***** end of loop to read netsid output data *****
c******************************************************
90    continue
c
      close (5)
C*****************************************************************
C  OUTPUT TO ROTEPLAN.OUT FILE
c
      open (unit=6,file='roteplan.out',status='new')
c
```

```
C  INFO ABOUT ARTIFICIAL FLOW VARIABLES
      nodnum=0
c
      if (k.GT.0) then
        write (6,620) k
620     format (' ','**** ',I3,' ARTIFICIAL VARIABLES PRESENT: ',
     $     'NO FEASIBLE SOLUTION  ****'/
     $     ' from  to    unit          to-node characteristics ',
     $     '   attrition'/
     $     ' node  node  cost  flow    year  duty   asd  flyacc',
     $     '   demanded'/
     $     ' ----  ----  ----  ----    ----  ----  ----  ------',
     $     '   --------')
      endif
c
      do 88 n=1,k
        tnode = INT(artif(n,2))
        do 93 time=1,5
          do 93 duty=1,4
            do 93 asdyg=0,18
              if (asdyg.lt.6) then
                minfly=asdyg
              else
                minfly=6
              endif
              if (asdyg.gt.11) then
                maxfly=11
              else
                maxfly=asdyg
              endif
              do 93 flycre=minfly,maxfly
                if (tnode.eq.node(time,duty,asdyg,
     $            flycre,0)) goto 50
93      continue
c
c  assign 'bogus' values just in case loop can't find an 'equal'.
c  If these appear on roteplan.out, need to look at nodearc.out
c  to determine actual node where artificial flow is occuring.
c
        time=8
        duty=5
        asdyg=19
        flycre=12
c
50      continue
        write (6,650) artif(n,1),artif(n,2),artif(n,4),artif(n,3),
     $    time,duty,asdyg,flycre,atr(time,duty,
     $    asdyg,flycre)
650     format (' ',F5.0,1x,F5.0,1x,F4.0,2x,F5.1,1x,4(I6),F10.1)
c
88    continue
c**********************************************************************
```

```
C  OPTIMAL SOLUTION ARC FLOWS CONVERTED TO ASSIGNMENT POLICY INFO
C
      do 91 time=1,5
        if (time.GT.1) then
          gtotal=tassign(1)+tassign(2)+tassign(3)+tassign(4)
          write (6,750) tassign(1),tassign(2),tassign(3),tassign(4),
     $      gtotal,tatrit
750       format (' '/,'  TOTAL',4X,2(1x,F6.1,4X),2x,F6.1,4x,F6.1,
     $      5x,2F6.1)
        endif
c
        tatrit=0
        do 80 i=1,4
          tassign(i)=0
80      continue
c
        write (6,700) fy(time)
700     format ('1'//,'  Optimal assignments for fiscal year ',I3//
     $    '  ASD  GATE   # to each duty and cost ',15x,
     $    '  TOTAL   CURR YR'/
     $    '  YEAR TIME   FLY   $    SUP   $    AFIT   $    ',
     $    'PME   $  ASSIGN  ATTRIT'/)
c
        do 91 asdyg=0,18
c
          if (asdyg.LT.6) then
            minfly=asdyg
          else
            minfly=6
          endif
c
          if (asdyg.GT.11) then
            maxfly=11
          else
            maxfly=asdyg
          endif
c
          do 91 flycre=minfly,maxfly
c
c  nodenu(i) are assigned the node numbers for the respective
c  duty types for a particular time/asd group/flycre combination.
c  Assign(end duty) holds total flow of all rotations into end duty
c  and cost(end duty) holds unit cost of rotations into end duty
c  for that particular time/asd group/flycre combo.
c
              nodenu(1)=node(time,1,asdyg,flycre,0)
              nodenu(2)=node(time,2,asdyg,flycre,0)
              nodenu(3)=node(time,3,asdyg,flycre,0)
              nodenu(4)=node(time,4,asdyg,flycre,0)
c
              tgroup=0
              do 85 enddut=1,4
```

```
                 assign(enddut)=nassign(nodenu(1),enddut,1)+
     $             nassign(nodenu(2),enddut,1)+
     $             nassign(nodenu(3),enddut,1)+
     $             nassign(nodenu(4),enddut,1)
                 cost(enddut)=AMAX1(nassign(nodenu(1),enddut,2),
     $             nassign(nodenu(2),enddut,2),
     $             nassign(nodenu(3),enddut,2),
     $             nassign(nodenu(4),enddut,2))
                 tgroup=tgroup+assign(enddut)
                 tassign(enddut)=tassign(enddut)+assign(enddut)
c
c  sum rotations by end duty and gate attained
c
                 if (flycre.lt.6.and.asdyg.lt.12)
     $gate0(time,enddut)=gate0(time,enddut)+assign(enddut)
                 if (flycre.ge.6.and.flycre.lt.9.and.
     $asdyg.lt.18) gate1y(time,enddut)=gate1y(time,enddut)+
     $assign(enddut)
                 if (flycre.lt.6.and.asdyg.ge.12)
     $gate1n(time,enddut)=gate1n(time,enddut)+assign(enddut)
                 if (flycre.ge.9.and.flycre.lt.11.and.
     $asdyg.lt.18) gate2y(time,enddut)=gate2y(time,enddut)+
     $assign(enddut)
                 if (flycre.ge.6.and.flycre.lt.9.and.
     $asdyg.ge.18) gate2n(time,enddut)=gate2n(time,enddut)+
     $assign(enddut)
                 if (flycre.ge.11.and.asdyg.le.18)
     $gate3y(time,enddut)=gate3y(time,enddut)+assign(enddut)
                 if (flycre.ge.9.and.flycre.lt.11.and.
     $asdyg.ge.18) gate3n(time,enddut)=gate3n(time,enddut)+
     $assign(enddut)
c
C  SUM ROTATIONS WITH COST (ROTEWC) BY TIME, END DUTY, ASD
C  GATE CREDIT.
C
                 IF(COST(ENDDUT).NE.0) THEN
                   ROTEWC(TIME,ENDDUT,ASDYG,FLYcre,1)=
     $               COST(ENDDUT)
                   ROTEWC(TIME,ENDDUT,ASDYG,FLYcre,2)=
     $               ROTEWC(TIME,ENDDUT,ASDYG,FLYcre,2)+
     $               ASSIGN(ENDDUT)
                   ROTEWC(TIME,ENDDUT,ASDYG,FLYcre,3)=
     $               ROTEWC(TIME,ENDDUT,ASDYG,FLYcre,3)+
     $               NASSIGN(NODENU(1),ENDDUT,1)
                   ROTEWC(TIME,ENDDUT,ASDYG,FLYcre,4)=
     $               ROTEWC(TIME,ENDDUT,ASDYG,FLYcre,4)+
     $               NASSIGN(NODENU(2),ENDDUT,1)
                   ROTEWC(TIME,ENDDUT,ASDYG,FLYcre,5)=
     $               ROTEWC(TIME,ENDDUT,ASDYG,FLYcre,5)+
     $               NASSIGN(NODENU(3),ENDDUT,1)
                   ROTEWC(TIME,ENDDUT,ASDYG,FLYcre,6)=
     $               ROTEWC(TIME,ENDDUT,ASDYG,FLYcre,6)+
```

```
     $                NASSIGN(NODENU(4),ENDDUT,i)
                    ENDIF
85                continue
C
C   CALCULATE ATTRITION FLOWS
C
                  natrit=atr(time,1,asdYG,flycre) +
     $              atr(time,2,asdyg,flycre) +
     $              atr(time,3,asdyg,flycre) +
     $              atr(time,4,asdyg,flycre)
                  tatrit=tatrit+natrit
C
C   WRITE TO OUTPUT FILE
C
                  write (6,800) asdyg,flycre,
     $ assign(1),cost(1),assign(2),cost(2),
     $ assign(3),cost(3),assign(4),cost(4),tgroup,natrit
C
800     format (' ',I3,I5,4x,2(F5.1,F4.0,2x),2(F6.1,F4.0),
     $   1x,2F6.1)
C*******************************************************************
C   CALCULATE PERCENTAGES ASSIGNED TO EACH DUTY AND ATTRITED
C   FOR EACH ASD GROUP/GATE CREDIT COMBINATION
C
                  total=tgroup+natrit
C
                  if (total.ne.0) then
                    passign(time,1,asdyg,flycre)=assign(1)/total
                    passign(time,2,asdyg,flycre)=assign(2)/total
                    passign(time,3,asdyg,flycre)=assign(3)/total
                    passign(time,4,asdyg,flycre)=assign(4)/total
                    passign(time,5,asdyg,flycre)=natrit/total
                  endif
C
91      continue
C
        gtotal=tassign(1)+tassign(2)+tassign(3)+tassign(4)
        write (6,750) tassign(1),tassign(2),tassign(3),tassign(4),
     $   gtotal,tatrit
C
        do 97 time=1,5
          write(6,850) fy(time)
850       format('1'//,'  optimal assignment percentages for fiscal ',
     $      'year ',I3//'  asd    fly       * percent to each duty * '/
     $      '  group credit   fly      sup      afit     pme      attrit',
     $      //)
C
          do 97 asdyg=0,18
C
            if (asdyg.lt.6) then
              minfly=asdyg
            else
```

```
            minfly=6
          endif
c
          if (asdyg.gt.11) then
            maxfly=11
          else
            maxfly=asdyg
          endif
c
          do 97 flycre=minfly,maxfly
            write(6,860) asdyg,flycre,
     $        passign(time,1,asdyg,flycre)*100.,
     $        passign(time,2,asdyg,flycre)*100.,
     $        passign(time,3,asdyg,flycre)*100.,
     $        passign(time,4,asdyg,flycre)*100.,
     $        passign(time,5,asdyg,flycre)*100.
860         format (' ',I5,3x,I4,4x,5(F5.1,3x))
c
            do 97 enddut=1,5
              passign(6,enddut,asdyg,flycre) =
     $          passign(6,enddut,asdyg,flycre) +
     $          passign(time,enddut,asdyg,flycre)
c
97    continue
c
C  OVERALL PERCENTAGES FOR ALL 5 YEARS
C
      write(6,870) fy(1),fy(5)
870   format ('1'//,' optimal assignment percentages averaged ',
     $ 'over ',I3,' thru ',I3//' asd   fly      * percent to ',
     $ 'each duty *'/' group credit   fly     sup     afit    ',
     $ 'pme     attrit'//)
c
      do 98 asdyg=0,18
c
        if (asdyg.lt.6) then
          minfly=asdyg
        else
          minfly=6
        endif
c
        if (asdyg.gt.11) then
          maxfly=11
        else
          maxfly=asdyg
        endif
c
        do 98 flycre=minfly,maxfly
          write(6,860) asdyg,flycre,
     $      passign(6,1,asdyg,flycre)*100./5.,
     $      passign(6,2,asdyg,flycre)*100./5.,
     $      passign(6,3,asdyg,flycre)*100./5.,
```

```
     $       passign(6,4,asdyg,flycre)*100./5.,
     $       passign(6,5,asdyg,flycre)*100./5.
c
98    continue
c
c*************************************************************
C  WRITE ASSIGNMENTS WITH AN ASSOCIATED COST
C
      WRITE(6,745)
745   FORMAT('1','ASSIGNMENTS THAT RESULT IN MISSED GATES '/
     $ ' ','       NEW          GATE  NUMBER    UNIT',
     $ '     DUTY ROTATING FROM:'/
     $ ' ','YEAR  DUTY  ASD   TIME  ASSIGNED  COST',
     $ '      FLY   SUP   AFIT  PME '/
     $ ' ','----  ----  ----  ----  --------  ----',
     $ '      ----  ----  ----  ----'/)
C
      DO 22 TIME=1,5
        WRITE(6,747)
747     FORMAT(' ')
        DO 22 I=1,4
          DO 22 J=0,18
            DO 22 K=0,11
              IF (ROTEWC(TIME,I,J,K,1).NE.0)
     $          WRITE(6,746) TIME,I,J,K,ROTEWC(TIME,I,J,K,2),
     $          ROTEWC(TIME,I,J,K,1),ROTEWC(TIME,I,J,K,3),
     $          ROTEWC(TIME,I,J,K,4),ROTEWC(TIME,I,J,K,5),
     $          ROTEWC(TIME,I,J,K,6)
746   FORMAT (' ',4(I4,2X),F7.1,3X,F3.0,6X,4(F4.0,2X))
22    CONTINUE
C*************************************************************
c  write assignment solution based on which gates have been
c  attained.
c
      write(6,760)
760   format('1','OPTIMAL ASSIGNMENT POLICY '/' broken out ',
     $    'by gate attained (hit) or missed'/' (hit or miss ',
     $    'status is as of the start of the assignment)'/
     $    ' GATE HIT OR MISSED:       hit  miss  hit  miss  ',
     $    'hit  miss'/
     $    ' year  duty          none   1    1     2    2    ',
     $    ' 3    3'/
     $    ' ----  ----          ----  ----  ----  ----  ----  ',
     $    '----  ----')
c
      do 21 time=1,5
        write(6,768)
768     format (' ')
        do 21 enddut=1,4
          write(6,765) time,enddut,gate0(time,enddut),
     $     gate1y(time,enddut),gate1n(time,enddut),
     $     gate2y(time,enddut),gate2n(time,enddut),
```

```
      $     gate3y(time,enddut),gate3n(time,enddut)
765   format(' ',I2,4x,I2,11x,F5.0,2x,3(F4.0,1x,F4.0,2x))
21    continue
      write(6,767)
767   format(/' ','NOTE:  duty 1 = fly, duty 2 = staff/supp,'/
      $  ' duty 3 = AFIT, duty 4 = PME/ASTRA')
      close (6)
c
      return
      end
c
```

Appendix C: Samples of Network Data Files

Sample data lines from FOR001.dat:

```
     2    229.00
     4     60.00
     5    281.00
     .       .
     .       .
   870     -1.00
   908    -53.00
   975  -3153.00
```

Sample data lines from FOR002.dat:

```
     1      1    586     -1.00       0.00
     2      2    587     -1.00       0.00
     3      3    588     -1.00       0.00
     .      .      .       .           .
     .      .      .       .           .
  2372    972    975     -1.00       0.00
  2373    973    975     -1.00       0.00
  2374    974    975     -1.00       0.00
```

Sample data lines from FOR003.dat:

```
     1      1          1.
     2      1          1.
     9      1          1.
     .      .           .
     .      .           .
    40   2370          1.
    69   2370          1.
    70   2370          1.
```

Sample data lines from FOR004.dat:

```
     1   1172.00L
     2    732.00G
     3    712.00L
     4    516.00G
     .       . .
     .       . .
    72    790.00G
    73   1096.00G
    74   1080.00G
    75   1096.00G
```

Appendix D: Sample of NETSID Output

Note: This appendix contains only a portion of the NETSID output file FOR007.dat. The actual file contains a complete listing of all solution arc flows.

```
N E T S I D

  ENTER REINVT  ITERATION    0
  ENTER REINVT  ITERATION  145
  ENTER REINVT  ITERATION  290
  ENTER REINVT  ITERATION  435
  ENTER REINVT  ITERATION  580
  ENTER REINVT  ITERATION  725
  ENTER REINVT  ITERATION  870
  ENTER REINVT  ITERATION 1015
  ENTER REINVT  ITERATION 1160
  ENTER REINVT  ITERATION 1305
  ENTER REINVT  ITERATION 1450
  ENTER REINVT  ITERATION 1595
  ENTER REINVT  ITERATION 1740
  ENTER REINVT  ITERATION 1885

THE FOLLOWING ROUTINES CHECK THE SOLUTION
FOR CONSISTENCY
  ENTER CHEKQI  ITERATION 1917
  ENTER FESCHK  ITERATION 1917
  ENTER DUALCK  ITERATION 1917

  OBJECTIVE FUNCTION VALUE  0.649000000E+03

  OPTIMAL SOLUTION AT ITERATION   1917

  BASIC VARIABLES --- ARTIFICIALS HAVE NAME - 99999
```

INDEX	NAME	FROM	TO	VALUE	COST
2	2	2	587	0.229000000E+03	0.000000000E+00
4	4	4	589	0.600000000E+02	0.000000000E+00
5	5	5	591	0.281000000E+03	0.000000000E+00
6	6	6	594	0.107000000E+03	0.000000000E+00
7	806	7	658	0.980000000E+02	0.000000000E+00
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
63	2298	708	970	0.800000000E+01	0.000000000E+00
67	1449	668	972	0.210000000E+02	0.000000000E+00
69	2307	785	974	0.500000000E+01	0.000000000E+00

Appendix E: GATES Output File SANITY.OUT

INPUT PARAMETERS

DUTY TYPE:	FLY	SUP	AFIT	PME
duty duration:	3	3	1	1

GATE COSTS:

GATE TIME	ASD	MISS COST
6	12	3
9	18	2
11	18	1
0	0	0
0	0	0
1	1	4
2	2	4
3	3	4
4	4	4
5	5	4
6	6	4
0	0	0

ATTRITION ADJUSTMENTS

YEAR	DUTY	ASD	FLY	ADJUSTMENT
1	1	9	8	3.
1	1	9	9	-3.
1	2	12	8	-1.
2	1	8	7	5.
2	1	8	8	-5.
2	1	9	8	5.
2	1	9	9	-5.
3	1	9	8	8.
3	1	9	9	-8.
3	1	12	10	-1.
3	1	12	11	1.
4	1	7	7	40.
4	1	8	8	2.
4	1	10	10	4.
5	1	8	8	-40.
5	1	9	9	-2.
5	1	11	11	-4.

ARRIVED-STATION STATISTICS
(number that arrived at initial duty station
during indicated time periods.
1st column = fy 1st half, 2nd column - fy 2nd half)

duty type	before FY84	FY84		FY85		FY86		FY87		total
1	407.	83	229.	251.	378.	270.	364.	326.	289.	2597.
2	96.	26.	86.	46.	121.	52.	145.	61.	107.	740.
3	0.	0.	0.	0.	3.	0.	15.	0.	12.	30.
4	0.	0.	0.	0.	1.	0.	1.	1.	39.	42.
total	503	109	315	297	503	322	525	388	447	

DUTY COMPOSITION STATISTICS

NOTE: these stats represent the situation after the model has forced certain duty, ASD, and gate credit combinations

(columns sum to 100 percent)
percent of total personnel in each duty type that belong to each asd group

asd group	fly	sup	afit	pme
0	0.	0.	0.	0.
1	1.	0.	0.	0.
2	6.	0.	0.	0.
3	6.	0.	0.	0.
4	10.	0.	0.	0.
5	10.	0.	0.	0.
6	11.	1.	3.	0.
7	7.	1.	17.	5.
8	5.	2.	17.	2.
9	2.	2.	30.	2.
10	2.	4.	10.	5.
11	2.	4.	10.	12.
12	2.	4.	3.	14.
13	2.	6.	7.	14.
14	3.	6.	0.	7.
15	3.	7.	3.	5.
16	3.	12.	0.	0.
17	5.	10.	0.	2.
18	22.	40.	0.	31.

(rows sum to 100 percent)
percent of total personnel in each asd group that are in each dutytype

asd group	fly	sup	afit	pme
0	0.	0.	0.	0.
1	100.	0.	0.	0.
2	100.	0.	0.	0.
3	100.	0.	0.	0.
4	100.	0.	0.	0.
5	100.	0.	0.	0.
6	98.	2.	0.	0.
7	93.	3.	3.	1.
8	87.	9.	4.	1.
9	69.	20.	10.	1.
10	58.	36.	4.	2.
11	57.	34.	3.	5.
12	52.	39.	1.	8.
13	53.	40.	2.	5.
14	61.	36.	0.	2.
15	57.	40.	1.	1.
16	48.	52.	0.	0.
17	61.	39.	0.	1.
18	65.	34.	0.	1.

ASD GROUP COMPOSITION STATISTICS
NOTE: These numbers are also after the model
has forced certain combinations

(rows sum to 100 percent)
% of each asd group by gate credit accumulated

asd group	6 yrs	7 yrs	8 yrs	9 yrs	10 yrs	11 yrs
0	0.	0.	0.	0.	0.	0.
1	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.
3	0.	0.	0.	0.	0.	0.
4	0.	0.	0.	0.	0.	0.
5	0.	0.	0.	0.	0.	0.
6	100.	0.	0.	0.	0.	0.
7	47.	53.	0.	0.	0.	0.
8	8.	49.	43.	0.	0.	0.
9	20.	18.	36.	27.	0.	0.
10	22.	13.	11.	29.	25.	0.
11	13.	11.	19.	13.	18.	27.
12	3.	8.	9.	24.	9.	47.
13	1.	3.	13.	17.	13.	53.
14	1.	1.	5.	12.	20.	61.
15	0.	1.	2.	15.	18.	64.
16	0.	0.	1.	16.	16.	67.
17	0.	0.	1.	10.	16.	74.
18	17.	0.	0.	6.	5.	72.

YEAR:	88	89	90	91	92

TOTAL ROTATIONS
grouped by 'from' duty and 'initial' asd group

DUTY 1; ASD 0	0.	0.	0.	0.	0.
DUTY 1; ASD 1	0.	1.	18.	0.	0.
DUTY 1; ASD 2	0.	49.	102.	0.	0.
DUTY 1; ASD 3	60.	82.	12.	0.	0.
DUTY 1; ASD 4	193.	5.	52.	0.	0.
DUTY 1; ASD 5	107.	36.	112.	0.	0.
DUTY 1; ASD 6	142.	83.	55.	0.	0.
DUTY 1; ASD 7	112.	42.	25.	0.	0.
DUTY 1; ASD 8	68.	33.	22.	0.	0.
DUTY 1; ASD 9	26.	23.	13.	0.	0.
DUTY 1; ASD 10	20.	17.	11.	0.	0.
DUTY 1; ASD 11	26.	15.	14.	0.	0.
DUTY 1; ASD 12	27.	8.	4.	0.	0.
DUTY 1; ASD 13	27.	21.	15.	0.	0.
DUTY 1; ASD 14	44.	19.	12.	0.	0.
DUTY 1; ASD 15	45.	25.	8.	0.	0.
DUTY 1; ASD 16	39.	24.	16.	0.	0.
DUTY 1; ASD 17	65.	23.	32.	0.	0.
DUTY 1; ASD 18	347.	128.	92.	0.	0.
DUTY 2; ASD 0	0.	0.	0.	0.	0.
DUTY 2; ASD 1	0.	0.	0.	0.	0.
DUTY 2; ASD 2	0.	0.	0.	0.	0.
DUTY 2; ASD 3	0.	0.	0.	0.	0.
DUTY 2; ASD 4	0.	0.	0.	0.	0.
DUTY 2; ASD 5	0.	0.	0.	0.	0.
DUTY 2; ASD 6	2.	0.	3.	0.	0.
DUTY 2; ASD 7	2.	0.	4.	0.	0.
DUTY 2; ASD 8	1.	9.	3.	0.	0.
DUTY 2; ASD 9	4.	7.	7.	0.	0.
DUTY 2; ASD 10	13.	8.	9.	0.	0.
DUTY 2; ASD 11	14.	9.	10.	0.	0.
DUTY 2; ASD 12	13.	12.	4.	0.	0.
DUTY 2; ASD 13	20.	13.	14.	0.	0.
DUTY 2; ASD 14	11.	21.	12.	0.	0.
DUTY 2; ASD 15	24.	17.	14.	0.	0.
DUTY 2; ASD 16	45.	21.	21.	0.	0.
DUTY 2; ASD 17	43.	13.	21.	0.	0.
DUTY 2; ASD 18	183.	67.	46.	0.	0.
DUTY 3; ASD 0	0.	0.	0.	0.	0.
DUTY 3; ASD 1	0.	0.	0.	0.	0.
DUTY 3; ASD 2	0.	0.	0.	0.	0.
DUTY 3; ASD 3	0.	0.	0.	0.	0.
DUTY 3; ASD 4	0.	0.	0.	0.	0.
DUTY 3; ASD 5	0.	0.	0.	0.	0.
DUTY 3; ASD 6	1.	0.	0.	0.	0.

DUTY 3; ASD 7	5.	0.	0.	0.	0.
DUTY 3; ASD 8	5.	0.	0.	0.	0.
DUTY 3; ASD 9	9.	0.	0.	0.	0.
DUTY 3; ASD 10	3.	0.	0.	0.	0.
DUTY 3; ASD 11	3.	0.	0.	0.	0.
DUTY 3; ASD 12	1.	0.	0.	0.	0.
DUTY 3; ASD 13	2.	0.	0.	0.	0.
DUTY 3; ASD 14	0.	0.	0.	0.	0.
DUTY 3; ASD 15	1.	0.	0.	0.	0.
DUTY 3; ASD 16	0.	0.	0.	0.	0.
DUTY 3; ASD 17	0.	0.	0.	0.	0.
DUTY 3; ASD 18	0.	0.	0.	0.	0.
DUTY 4; ASD 0	0.	0.	0.	0.	0.
DUTY 4; ASD 1	0.	0.	0.	0.	0.
DUTY 4; ASD 2	0.	0.	0.	0.	0.
DUTY 4; ASD 3	0.	0.	0.	0.	0.
DUTY 4; ASD 4	0.	0.	0.	0.	0.
DUTY 4; ASD 5	0.	0.	0.	0.	0.
DUTY 4; ASD 6	0.	0.	0.	0.	0.
DUTY 4; ASD 7	2.	0.	0.	0.	0.
DUTY 4; ASD 8	1.	0.	0.	0.	0.
DUTY 4; ASD 9	1.	0.	0.	0.	0.
DUTY 4; ASD 10	2.	0.	0.	0.	0.
DUTY 4; ASD 11	5.	0.	0.	0.	0.
DUTY 4; ASD 12	6.	0.	0.	0.	0.
DUTY 4; ASD 13	6.	0.	0.	0.	0.
DUTY 4; ASD 14	3.	0.	0.	0.	0.
DUTY 4; ASD 15	2.	0.	0.	0.	0.
DUTY 4; ASD 16	0.	0.	0.	0.	0.
DUTY 4; ASD 17	1.	0.	0.	0.	0.
DUTY 4; ASD 18	13.	0.	0.	0.	0.

ROTATIONS THAT HAVE MISSED THEIR GATES

YEAR:	88	89	90	91	92
1ST GATE	0	0	0	0	0
2ND GATE	41	63	49	0	0
3RD GATE	54	25	28	0	0

YEAR:	88	89	90	91	92
AVAILABLES					
GAINS					
UFT GAINS:	229	248	248	235	235
FAIP GAINS:	88	67	67	64	64
TOTAL GAINS:	317.	315.	315.	299.	299.
GAINS TO DATE:	317.	632.	947.	1246.	1545.
ROTATIONS					
DUTY 1:	1348.	634.	615.	0.	0.
DUTY 2:	375.	197.	168.	0.	0.
DUTY 3:	30.	0.	0.	0.	0.
DUTY 4:	42.	0.	0.	0.	0.
TOTAL ROTES:	1795.	831.	783.	0.	0.
ROTES TO DATE:	1795.	2626.	3409.	3409.	3409.
ATTRITION:	271.	467.	444.	373.	246.
ATRIT TO DATE:	271.	738.	1182.	1555.	1801.
TOT AVAILABLE:	1841.	2520.	3174.	3100.	3153.

tot available = gains to date + rotations to date - attrition to date

YEAR:		88	89	90	91	92
REQUIREMENTS						
BASELINE REQUIREMENTS (from R.M.D.)						
DUTY	1:	2201.	2195.	2192.	2159.	2191.
DUTY	2:	979.	977.	1041.	1013.	1192.
DUTY	3:	31.	31.	33.	32.	38.
DUTY	4:	31.	31.	34.	33.	38.
TOTAL RQMTS:		3242.	3234.	3300.	3237.	3459.
UNDERMANNED (MIN) REQUIREMENTS						
DUTY	1:	1981.	1976.	1973.	1943.	1972.
DUTY	2:	881.	879.	937.	912.	1073.
DUTY	3:	29.	29.	31.	30.	36.
DUTY	4:	31.	31.	34.	33.	38.
TOTAL UNDER:		2922.	2915.	2975.	2918.	3119.
OVERMANNED (MAX) REQUIREMENTS						
DUTY	1:	2421.	2415.	2411.	2375.	2410.
DUTY	2:	1077.	1075.	1145.	1114.	1311.
DUTY	3:	33.	33.	35.	34.	40.
DUTY	4:	31.	31.	34.	33.	38.
TOTAL OVER:		3562.	3554.	3625.	3556.	3799.
ADJUSTMENT (FUTURE ROTATIONS)						
DUTY	1:	1249.	615.	0.	0.	0.
DUTY	2:	365.	168.	0.	0.	0.
DUTY	3:	0.	0.	0.	0.	0.
DUTY	4:	0.	0.	0.	0.	0.
TOTAL ADJUST:		1614.	783.	0.	0.	0.
ADJUSTED UNDERMANNED REQUIREMENTS (undermanned rqmts reduced for downstream rotations)						
DUTY	1:	732.	1361.	1973.	1943.	1972.
DUTY	2:	516.	711.	937.	912.	1073.
DUTY	3:	29.	29.	31.	30.	36.
DUTY	4:	31.	31.	34.	33.	38.
ADJUSTED OVERMANNED REQUIREMENTS (overmanned rqmts reduced for downstream rotations)						
DUTY	1:	1172.	1800.	2411.	2375.	2410.
DUTY	2:	712.	907.	1145.	1114.	1311.
DUTY	3:	33.	33.	35.	34.	40.
DUTY	4:	31.	31.	34.	33.	38.

ADJUSTED TOTAL REQUIREMENTS
(Requirements reduced for downstream rotations)

UNDERMANNED:	1308.	2132.	2975.	2918.	3119.
BASELINE:	1628.	2451.	3300.	3237.	3459.
OVERMANNED:	1948.	2771.	3625.	3556.	3799.

tot available corresponds to the flows in the network for each year and should fall between undermanned and overmanned adjusted total requirements.

ATTRITION

asd group	gate time	year 1 fly	year 1 sup	year 2 fly	year 2 sup	year 3 fly	year 3 sup	year 4 fly	year 4 sup	year 5 fly	year 5 sup
6	6	17.	0.	15.	0.	20.	0.	13.	0.	13.	0.
7	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
7	7	33.	0.	46.	0.	41.	0.	94.	0.	36.	0.
8	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
8	7	0.	0.	5.	0.	0.	0.	0.	0.	0.	0.
8	8	24.	0.	22.	0.	38.	0.	36.	0.	5.	0.
9	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
9	7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
9	8	3.	0.	5.	0.	8.	0.	0.	0.	0.	0.
9	9	10.	0.	12.	0.	10.	0.	26.	0.	21.	0.
10	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
10	7	0.	2.	0.	2.	0.	2.	0.	2.	0.	3.
10	8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
10	9	2.	0.	2.	0.	2.	0.	2.	0.	3.	0.
10	10	8.	0.	7.	0.	10.	0.	15.	0.	15.	0.
11	6	0.	1.	0.	1.	0.	1.	0.	1.	0.	1.
11	7	0.	1.	0.	1.	0.	1.	0.	1.	0.	1.
11	8	0.	3.	0.	2.	0.	2.	0.	2.	0.	3.
11	9	1.	0.	1.	0.	1.	0.	1.	0.	1.	0.
11	10	3.	0.	3.	0.	3.	0.	3.	0.	4.	0.
11	11	3.	0.	3.	0.	3.	0.	3.	0.	0.	0.
12	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
12	7	0.	1.	0.	1.	0.	0.	0.	0.	0.	1.
12	8	0.	0.	0.	1.	0.	0.	0.	0.	0.	1.
12	9	0.	1.	0.	1.	0.	1.	0.	1.	0.	1.
12	10	1.	[illegible]	1.	0.	0.	0.	1.	0.	1.	0.
12	11	3.	0.	3.	0.	3.	0.	2.	0.	3.	0.
13	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
13	7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
13	8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
13	9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
13	10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
13	11	1.	0.	1.	0.	1.	0.	1.	0.	1.	0.
14	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

14	7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
14	8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
14	9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
14	10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
14	11	0.	0.	1.	0.	0.	0.	0.	0.	0.	0.
15	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
15	7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
15	8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
15	9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
15	10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
15	11	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
16	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
16	7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
16	8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
16	9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
16	10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
16	11	1.	1.	1.	1.	1.	1.	0.	0.	0.	0.
17	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
17	7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
17	8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
17	9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
17	10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
17	11	1.	1.	0.	0.	0.	0.	0.	0.	0.	0.
18	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
18	7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
18	8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
18	9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
18	10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
18	11	89.	60.	197.	132.	177.	118.	101.	68.	79.	53.

Appendix F: GATES Output File ROTEPLAN.OUT

Optimal assignments for fiscal year 88

ASD YEAR	GATE TIME	# to each duty and cost FLY	$	SUP	$	AFIT	$	PME	$	TOTAL ASSIGN	CURR YR ATTRIT
0	0	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
1	1	229.0	0.	0.0	0.	0.0	0.	0.0	0.	229.0	0.0
2	2	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
3	3	60.0	0.	0.0	0.	0.0	0.	0.0	0.	60.0	0.0
4	4	281.0	0.	0.0	0.	0.0	0.	0.0	0.	281.0	0.0
5	5	107.0	0.	0.0	0.	0.0	0.	0.0	0.	107.0	0.0
6	6	26.0	0.	99.0	0.	0.0	0.	3.0	0.	128.0	17.0
7	6	29.0	0.	6.0	0.	18.0	0.	2.0	0.	55.0	0.0
7	7	15.0	0.	18.0	0.	0.0	0.	0.0	0.	33.0	33.0
8	6	1.0	0.	2.0	0.	0.0	0.	0.0	0.	3.0	0.0
8	7	3.0	0.	22.0	0.	10.0	0.	0.0	0.	35.0	0.0
8	8	3.0	0.	10.0	0.	0.0	0.	0.0	0.	13.0	24.0
9	6	0.0	0.	7.0	0.	0.0	0.	0.0	0.	7.0	0.0
9	7	1.0	0.	5.0	0.	0.0	0.	0.0	0.	6.0	0.0
9	8	2.0	0.	11.0	0.	0.0	0.	0.0	0.	13.0	3.0
9	9	0.0	0.	1.0	0.	0.0	0.	0.0	0.	1.0	10.0
10	6	0.0	0.	9.0	0.	1.0	0.	0.0	0.	10.0	0.0
10	7	0.0	0.	5.0	0.	0.0	0.	0.0	0.	5.0	2.0
10	8	1.0	0.	1.0	0.	0.0	0.	0.0	0.	2.0	0.0
10	9	6.0	0.	2.0	0.	0.0	0.	0.0	0.	8.0	2.0
10	10	1.0	0.	0.0	0.	0.0	0.	0.0	0.	1.0	8.0
11	6	0.0	0.	5.0	0.	0.0	0.	0.0	0.	5.0	1.0
11	7	0.0	0.	2.0	0.	0.0	0.	0.0	0.	2.0	1.0
11	8	0.0	0.	7.0	0.	0.0	0.	0.0	0.	7.0	3.0
11	9	0.0	0.	6.0	0.	0.0	0.	0.0	0.	6.0	1.0
11	10	0.0	0.	6.0	0.	0.0	0.	0.0	0.	6.0	3.0
11	11	0.0	0.	10.0	0.	0.0	0.	0.0	0.	10.0	3.0
12	6	1.0	0.	0.0	0.	0.0	0.	0.0	0.	1.0	0.0
12	7	3.0	0.	0.0	0.	0.0	0.	0.0	0.	3.0	1.0
12	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
12	9	0.0	0.	11.0	0.	0.0	0.	0.0	0.	11.0	1.0
12	10	0.0	0.	4.0	0.	0.0	0.	0.0	0.	4.0	1.0
12	11	0.0	0.	18.0	0.	4.0	0.	0.0	0.	22.0	3.0
13	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
13	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
13	8	0.0	0.	6.0	0.	0.0	0.	0.0	0.	6.0	0.0
13	9	0.0	0.	6.0	0.	0.0	0.	0.0	0.	6.0	0.0
13	10	0.0	0.	10.0	0.	0.0	0.	0.0	0.	10.0	0.0
13	11	0.0	0.	32.0	0.	0.0	0.	0.0	0.	32.0	1.0
14	6	1.0	0.	0.0	0.	0.0	0.	0.0	0.	1.0	0.0
14	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
14	8	0.0	0.	3.0	0.	0.0	0.	1.0	0.	4.0	0.0

14	9	0.0	0.	8.0	0.	0.0	0.	0.0	0.	8.0	0.0
14	10	0.0	0.	8.0	0.	0.0	0.	0.0	0.	8.0	0.0
14	11	0.0	0.	20.0	0.	0.0	0.	17.0	0.	37.0	0.0
15	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
15	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
15	8	2.0	0.	0.0	0.	0.0	0.	0.0	0.	2.0	0.0
15	9	16.0	0.	0.0	0.	0.0	0.	0.0	0.	16.0	0.0
15	10	12.0	0.	0.0	0.	0.0	0.	0.0	0.	12.0	0.0
15	11	33.0	0.	9.0	0.	0.0	0.	0.0	0.	42.0	0.0
16	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
16	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
16	8	1.0	0.	0.0	0.	0.0	0.	0.0	0.	1.0	0.0
16	9	11.0	0.	0.0	0.	0.0	0.	0.0	0.	11.0	0.0
16	10	11.0	0.	0.0	0.	0.0	0.	0.0	0.	11.0	0.0
16	11	27.0	0.	32.0	0.	0.0	0.	0.0	0.	59.0	2.0
17	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
17	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
17	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
17	9	9.0	0.	0.0	0.	0.0	0.	0.0	0.	9.0	0.0
17	10	18.0	0.	0.0	0.	0.0	0.	0.0	0.	18.0	0.0
17	11	18.0	0.	54.0	0.	0.0	0.	8.0	0.	80.0	2.0
18	6	40.0	2.	0.0	0.	0.0	0.	0.0	0.	40.0	0.0
18	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
18	8	1.0	1.	0.0	0.	0.0	0.	0.0	0.	1.0	0.0
18	9	29.0	1.	0.0	0.	0.0	0.	0.0	0.	29.0	0.0
18	10	25.0	0.	0.0	0.	0.0	0.	0.0	0.	25.0	0.0
18	11	149.0	0.	150.0	0.	0.0	0.	0.0	0.	299.0	149.0
TOTAL		1172.0		605.0		33.0		31.0		1841.0	271.0

Optimal assignments for fiscal year 89

ASD YEAR	GATE TIME	# to each duty and cost FLY	$	SUP	$	AFIT	$	PME	$	TOTAL ASSIGN	CURR YR ATTRIT
0	0	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
1	1	248.0	0.	0.0	0.	0.0	0.	0.0	0.	248.0	0.0
2	2	1.0	0.	0.0	0.	0.0	0.	0.0	0.	1.0	0.0
3	3	49.0	0.	0.0	0.	0.0	0.	0.0	0.	49.0	0.0
4	4	149.0	0.	0.0	0.	0.0	0.	0.0	0.	149.0	0.0
5	5	5.0	0.	0.0	0.	0.0	0.	0.0	0.	5.0	0.0
6	6	21.0	0.	0.0	0.	0.0	0.	0.0	0.	21.0	15.0
7	6	3.0	0.	0.0	0.	0.0	0.	0.0	0.	3.0	0.0
7	7	15.0	0.	17.0	0.	0.0	0.	5.0	0.	37.0	46.0
8	6	1.0	0.	19.0	0.	0.0	0.	0.0	0.	20.0	0.0
8	7	4.0	0.	6.0	0.	2.0	0.	0.0	0.	12.0	5.0
8	8	0.0	0.	3.0	0.	0.0	0.	0.0	0.	3.0	22.0
9	6	0.0	0.	0.0	0.	5.0	0.	0.0	0.	5.0	0.0
9	7	1.0	0.	15.0	0.	0.0	0.	0.0	0.	16.0	0.0
9	8	3.0	0.	10.0	0.	0.0	0.	0.0	0.	13.0	5.0
9	9	0.0	0.	1.0	0.	0.0	0.	0.0	0.	1.0	12.0
10	6	5.0	0.	0.0	0.	0.0	0.	0.0	0.	5.0	0.0
10	7	0.0	0.	3.0	0.	0.0	0.	0.0	0.	3.0	2.0
10	8	0.0	0.	3.0	0.	0.0	0.	0.0	0.	3.0	0.0
10	9	4.0	0.	0.0	0.	2.0	0.	0.0	0.	6.0	2.0
10	10	0.0	0.	2.0	0.	0.0	0.	0.0	0.	2.0	7.0
11	6	0.0	0.	1.0	0.	3.0	0.	0.0	0.	4.0	1.0
11	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	1.0
11	8	0.0	0.	0.0	0.	1.0	0.	0.0	0.	1.0	2.0
11	9	0.0	0.	0.0	0.	2.0	0.	0.0	0.	2.0	1.0
11	10	0.0	0.	0.0	0.	4.0	0.	0.0	0.	4.0	3.0
11	11	0.0	0.	0.0	0.	4.0	0.	0.0	0.	4.0	3.0
12	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
12	7	0.0	0.	0.0	0.	6.0	0.	0.0	0.	6.0	1.0
12	8	1.0	0.	0.0	0.	0.0	0.	0.0	0.	1.0	1.0
12	9	0.0	0.	0.0	0.	4.0	0.	0.0	0.	4.0	1.0
12	10	2.0	0.	0.0	0.	0.0	0.	0.0	0.	2.0	1.0
12	11	0.0	0.	4.0	0.	0.0	0.	0.0	0.	4.0	3.0
13	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
13	7	0.0	0.	0.0	0.	0.0	0.	1.0	0.	1.0	0.0
13	8	0.0	0.	1.0	0.	0.0	0.	2.0	0.	3.0	0.0
13	9	0.0	0.	0.0	0.	0.0	0.	7.0	0.	7.0	0.0
13	10	0.0	0.	0.0	0.	0.0	0.	4.0	0.	4.0	0.0
13	11	0.0	0.	4.0	0.	0.0	0.	4.0	0.	8.0	1.0
14	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
14	7	1.0	0.	0.0	0.	0.0	0.	0.0	0.	1.0	0.0
14	8	1.0	0.	2.0	0.	0.0	0.	0.0	0.	3.0	0.0
14	9	7.0	0.	0.0	0.	0.0	0.	0.0	0.	7.0	0.0
14	10	6.0	0.	0.0	0.	0.0	0.	0.0	0.	6.0	0.0
14	11	6.0	0.	10.0	0.	0.0	0.	0.0	0.	16.0	1.0
15	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
15	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0

15	8	3.0	0.	0.0	0.	0.0	0.	0.0	0.	3.0	0.0
15	9	5.0	0.	0.0	0.	0.0	0.	0.0	0.	5.0	0.0
15	10	4.0	0.	0.0	0.	0.0	0.	0.0	0.	4.0	0.0
15	11	0.0	0.	46.0	0.	0.0	0.	0.0	0.	46.0	0.0
16	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
16	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
16	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
16	9	4.0	0.	0.0	0.	0.0	0.	0.0	0.	4.0	0.0
16	10	5.0	0.	0.0	0.	0.0	0.	0.0	0.	5.0	0.0
16	11	0.0	0.	31.0	0.	0.0	0.	0.0	0.	31.0	2.0
17	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
17	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
17	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
17	9	5.0	0.	0.0	0.	0.0	0.	0.0	0.	5.0	0.0
17	10	8.0	0.	0.0	0.	0.0	0.	0.0	0.	8.0	0.0
17	11	12.0	0.	12.0	0.	0.0	0.	8.0	0.	32.0	0.0
18	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
18	7	103.0	2.	0.0	0.	0.0	0.	0.0	0.	103.0	0.0
18	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
18	9	15.0	1.	0.0	0.	0.0	0.	0.0	0.	15.0	0.0
18	10	40.0	0.	0.0	0.	0.0	0.	0.0	0.	40.0	0.0
18	11	50.0	0.	96.0	0.	0.0	0.	0.0	0.	146.0	329.0
TOTAL		787.0		286.0		33.0		31.0		1137.0	467.0

Optimal assignments for fiscal year 90

ASD YEAR	GATE TIME	# to each duty and cost FLY	$	SUP	$	AFIT	$	PME	$	TOTAL ASSIGN	CURR YR ATTRIT
0	0	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
1	1	248.0	0.	0.0	0.	0.0	0.	0.0	0.	248.0	0.0
2	2	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
3	3	18.0	0.	0.0	0.	0.0	0.	0.0	0.	18.0	0.0
4	4	169.0	0.	0.0	0.	0.0	0.	0.0	0.	169.0	0.0
5	5	12.0	0.	0.0	0.	0.0	0.	0.0	0.	12.0	0.0
6	6	0.0	0.	6.0	0.	21.0	0.	5.0	0.	32.0	20.0
7	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
7	7	0.0	0.	71.0	0.	0.0	0.	0.0	0.	71.0	41.0
8	6	0.0	0.	3.0	0.	0.0	0.	0.0	0.	3.0	0.0
8	7	0.0	0.	5.0	0.	0.0	0.	0.0	0.	5.0	0.0
8	8	0.0	0.	17.0	0.	0.0	0.	0.0	0.	17.0	38.0
9	6	0.0	0.	3.0	0.	0.0	0.	0.0	0.	3.0	0.0
9	7	0.0	0.	3.0	0.	0.0	0.	0.0	0.	3.0	0.0
9	8	0.0	0.	7.0	0.	0.0	0.	0.0	0.	7.0	8.0
9	9	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	10.0
10	6	0.0	0.	5.0	0.	0.0	0.	0.0	0.	5.0	0.0
10	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	2.0
10	8	0.0	0.	2.0	0.	0.0	0.	0.0	0.	2.0	0.0
10	9	0.0	0.	9.0	0.	0.0	0.	0.0	0.	9.0	2.0
10	10	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	10.0
11	6	0.0	0.	1.0	0.	0.0	0.	0.0	0.	1.0	1.0
11	7	0.0	0.	2.0	0.	0.0	0.	0.0	0.	2.0	1.0
11	8	0.0	0.	0.0	0.	1.0	0.	0.0	0.	1.0	2.0
11	9	0.0	0.	2.0	0.	1.0	0.	0.0	0.	3.0	1.0
11	10	0.0	0.	0.0	0.	3.0	0.	0.0	0.	3.0	3.0
11	11	0.0	0.	1.0	0.	0.0	0.	0.0	0.	1.0	3.0
12	6	0.0	0.	3.0	0.	2.0	0.	0.0	0.	5.0	0.0
12	7	0.0	0.	0.0	0.	0.0	0.	3.0	0.	3.0	0.0
12	8	0.0	0.	3.0	0.	1.0	0.	0.0	0.	4.0	0.0
12	9	0.0	0.	2.0	0.	2.0	0.	0.0	0.	4.0	1.0
12	10	0.0	0.	4.0	0.	0.0	0.	0.0	0.	4.0	0.0
12	11	0.0	0.	10.0	0.	0.0	0.	0.0	0.	10.0	3.0
13	6	0.0	0.	2.0	0.	0.0	0.	0.0	0.	2.0	0.0
13	7	0.0	0.	6.0	0.	0.0	0.	2.0	0.	8.0	0.0
13	8	0.0	0.	3.0	0.	0.0	0.	0.0	0.	3.0	0.0
13	9	0.0	0.	4.0	0.	0.0	0.	4.0	0.	8.0	0.0
13	10	0.0	0.	1.0	0.	0.0	0.	0.0	0.	1.0	0.0
13	11	0.0	0.	1.0	0.	0.0	0.	10.0	0.	11.0	1.0
14	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
14	7	0.0	0.	1.0	0.	0.0	0.	0.0	0.	1.0	0.0
14	8	0.0	0.	4.0	0.	0.0	0.	0.0	0.	4.0	0.0
14	9	0.0	0.	8.0	0.	0.0	0.	0.0	0.	8.0	0.0
14	10	0.0	0.	4.0	0.	0.0	0.	0.0	0.	4.0	0.0
14	11	0.0	0.	8.0	0.	0.0	0.	1.0	0.	9.0	0.0
15	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
15	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0

15	8	2.0	0.	0.0	0.	0.0	0.	0.0	0.	2.0	0.0
15	9	5.0	0.	0.0	0.	0.0	0.	0.0	0.	5.0	0.0
15	10	4.0	0.	0.0	0.	0.0	0.	0.0	0.	4.0	0.0
15	11	0.0	0.	18.0	0.	0.0	0.	0.0	0.	18.0	0.0
16	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
16	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
16	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
16	9	2.0	0.	0.0	0.	0.0	0.	0.0	0.	2.0	0.0
16	10	5.0	0.	0.0	0.	0.0	0.	0.0	0.	5.0	0.0
16	11	0.0	0.	15.0	0.	0.0	0.	0.0	0.	15.0	2.0
17	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
17	7	1.0	1.	0.0	0.	0.0	0.	0.0	0.	1.0	0.0
17	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
17	9	1.0	0.	0.0	0.	0.0	0.	0.0	0.	1.0	0.0
17	10	3.0	0.	0.0	0.	0.0	0.	0.0	0.	3.0	0.0
17	11	0.0	0.	9.0	0.	0.0	0.	8.0	0.	17.0	0.0
18	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
18	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
18	8	152.0	1.	0.0	0.	0.0	0.	0.0	0.	152.0	0.0
18	9	13.0	1.	0.0	0.	0.0	0.	0.0	0.	13.0	0.0
18	10	30.0	0.	0.0	0.	0.0	0.	0.0	0.	30.0	0.0
18	11	0.0	0.	49.0	0.	0.0	0.	1.0	0.	50.0	295.0
TOTAL		665.0		292.0		31.0		34.0		1022.0	444.0

Optimal assignments for fiscal year 91

ASD YEAR	GATE TIME	# to each duty and cost FLY	$	SUP	$	AFIT	$	PME	$	TOTAL ASSIGN	CURR YR ATTRIT
0	0	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
1	1	235.0	0.	0.0	0.	0.0	0.	0.0	0.	235.0	0.0
2	2	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
3	3	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
4	4	293.0	0.	0.0	0.	0.0	0.	0.0	0.	293.0	0.0
5	5	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
6	6	0.0	0.	47.0	0.	0.0	0.	0.0	0.	47.0	13.0
7	6	0.0	0.	26.0	0.	0.0	0.	0.0	0.	26.0	0.0
7	7	0.0	0.	182.0	0.	0.0	0.	5.0	0.	187.0	94.0
8	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
8	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
8	8	0.0	0.	71.0	0.	0.0	0.	0.0	0.	71.0	36.0
9	6	65.0	0.	0.0	0.	34.0	0.	0.0	0.	99.0	0.0
9	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
9	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
9	9	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	26.0
10	6	0.0	0.	6.0	0.	0.0	0.	0.0	0.	6.0	0.0
10	7	16.0	0.	0.0	0.	0.0	0.	0.0	0.	16.0	2.0
10	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
10	9	0.0	0.	27.0	0.	0.0	0.	0.0	0.	27.0	2.0
10	10	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	15.0
11	6	0.0	0.	1.0	0.	0.0	0.	0.0	0.	1.0	1.0
11	7	0.0	0.	21.0	0.	0.0	0.	0.0	0.	21.0	1.0
11	8	0.0	0.	8.0	0.	0.0	0.	0.0	0.	8.0	2.0
11	9	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	1.0
11	10	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	3.0
11	11	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	3.0
12	6	0.0	0.	7.0	0.	0.0	0.	0.0	0.	7.0	0.0
12	7	0.0	0.	5.0	0.	0.0	0.	0.0	0.	5.0	0.0
12	8	0.0	0.	1.0	0.	0.0	0.	11.0	0.	12.0	0.0
12	9	0.0	0.	1.0	0.	0.0	0.	0.0	0.	1.0	1.0
12	10	0.0	0.	3.0	0.	0.0	0.	0.0	0.	3.0	1.0
12	11	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	2.0
13	6	0.0	0.	2.0	0.	0.0	0.	9.0	0.	11.0	0.0
13	7	0.0	0.	8.0	0.	0.0	0.	0.0	0.	8.0	0.0
13	8	0.0	0.	2.0	0.	0.0	0.	0.0	0.	2.0	0.0
13	9	0.0	0.	4.0	0.	0.0	0.	0.0	0.	4.0	0.0
13	10	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
13	11	0.0	0.	7.0	0.	0.0	0.	0.0	0.	7.0	1.0
14	6	0.0	0.	5.0	0.	0.0	0.	0.0	0.	5.0	0.0
14	7	0.0	0.	4.0	0.	0.0	0.	0.0	0.	4.0	0.0
14	8	0.0	0.	7.0	0.	0.0	0.	0.0	0.	7.0	0.0
14	9	0.0	0.	10.0	0.	0.0	0.	0.0	0.	10.0	0.0
14	10	0.0	0.	6.0	0.	0.0	0.	0.0	0.	6.0	0.0
14	11	0.0	0.	20.0	0.	0.0	0.	0.0	0.	20.0	0.0
15	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
15	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0

15	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
15	9	12.0	0.	0.0	0.	0.0	0.	0.0	0.	12.0	0.0
15	10	7.0	0.	0.0	0.	0.0	0.	0.0	0.	7.0	0.0
15	11	0.0	0.	19.0	0.	0.0	0.	0.0	0.	19.0	0.0
16	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
16	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
16	8	6.0	0.	0.0	0.	0.0	0.	0.0	0.	6.0	0.0
16	9	6.0	0.	0.0	0.	0.0	0.	0.0	0.	6.0	0.0
16	10	10.0	0.	0.0	0.	0.0	0.	0.0	0.	10.0	0.0
16	11	27.0	0.	5.0	0.	0.0	0.	0.0	0.	32.0	0.0
17	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
17	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
17	8	3.0	0.	0.0	0.	0.0	0.	0.0	0.	3.0	0.0
17	9	9.0	0.	0.0	0.	0.0	0.	0.0	0.	9.0	0.0
17	10	8.0	0.	0.0	0.	0.0	0.	0.0	0.	8.0	0.0
17	11	12.0	0.	0.0	0.	0.0	0.	8.0	0.	20.0	0.0
18	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
18	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
18	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
18	9	152.0	1.	0.0	0.	0.0	0.	0.0	0.	152.0	0.0
18	10	13.0	0.	0.0	0.	0.0	0.	0.0	0.	13.0	0.0
18	11	20.0	0.	152.0	0.	0.0	0.	0.0	0.	172.0	169.0
TOTAL		894.0		657.0		34.0		33.0		1618.0	373.0

Optimal assignments for fiscal year 92

ASD YEAR	GATE TIME	# to each duty and cost FLY	$	SUP	$	AFIT	$	PME	$	TOTAL ASSIGN	CURR YR ATTRIT
0	0	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
1	1	235.0	0.	0.0	0.	0.0	0.	0.0	0.	235.0	0.0
2	2	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
3	3	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
4	4	312.0	0.	0.0	0.	0.0	0.	0.0	0.	312.0	0.0
5	5	1.0	0.	0.0	0.	0.0	0.	0.0	0.	1.0	0.0
6	6	0.0	0.	36.0	0.	0.0	0.	0.0	0.	36.0	13.0
7	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
7	7	89.0	0.	19.0	0.	0.0	0.	5.0	0.	113.0	36.0
8	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
8	7	0.0	0.	0.0	0.	5.0	0.	0.0	0.	5.0	0.0
8	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	5.0
9	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
9	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
9	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
9	9	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	21.0
10	6	0.0	0.	34.0	0.	0.0	0.	0.0	0.	34.0	0.0
10	7	0.0	0.	1.0	0.	13.0	0.	0.0	0.	14.0	3.0
10	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
10	9	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	3.0
10	10	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	15.0
11	6	0.0	0.	0.0	0.	18.0	0.	0.0	0.	18.0	1.0
11	7	0.0	0.	5.0	0.	0.0	0.	0.0	0.	5.0	1.0
11	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	3.0
11	9	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	1.0
11	10	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	4.0
11	11	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
12	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
12	7	0.0	0.	0.0	0.	0.0	0.	14.0	0.	14.0	1.0
12	8	0.0	0.	0.0	0.	0.0	0.	9.0	0.	9.0	1.0
12	9	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	1.0
12	10	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	1.0
12	11	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	3.0
13	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
13	7	0.0	0.	3.0	0.	0.0	0.	0.0	0.	3.0	0.0
13	8	0.0	0.	14.0	0.	0.0	0.	0.0	0.	14.0	0.0
13	9	0.0	0.	5.0	0.	0.0	0.	0.0	0.	5.0	0.0
13	10	0.0	0.	2.0	0.	0.0	0.	0.0	0.	2.0	0.0
13	11	0.0	0.	3.0	0.	0.0	0.	0.0	0.	3.0	1.0
14	6	0.0	0.	10.0	0.	0.0	0.	0.0	0.	10.0	0.0
14	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
14	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
14	9	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
14	10	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
14	11	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
15	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
15	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0

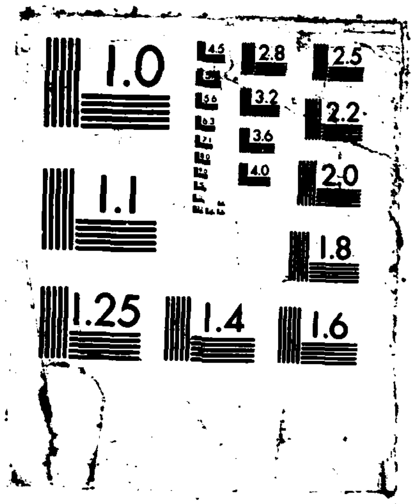
15	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
15	9	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
15	10	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
15	11	0.0	0.	7.0	0.	0.0	0.	0.0	0.	7.0	0.0
16	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
16	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
16	8	1.0	0.	0.0	0.	0.0	0.	0.0	0.	1.0	0.0
16	9	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
16	10	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
16	11	0.0	0.	4.0	0.	0.0	0.	0.0	0.	4.0	0.0
17	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
17	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
17	8	2.0	0.	0.0	0.	0.0	0.	0.0	0.	2.0	0.0
17	9	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
17	10	1.0	0.	0.0	0.	0.0	0.	0.0	0.	1.0	0.0
17	11	0.0	0.	30.0	0.	0.0	0.	0.0	0.	30.0	0.0
18	6	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
18	7	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
18	8	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
18	9	0.0	0.	0.0	0.	0.0	0.	0.0	0.	0.0	0.0
18	10	152.0	0.	0.0	0.	0.0	0.	0.0	0.	152.0	0.0
18	11	0.0	0.	186.0	0.	0.0	0.	10.0	0.	196.0	132.0
TOTAL		793.0		359.0		36.0		38.0		1226.0	246.0

optimal assignment percentages for fiscal year 88

asd group	fly credit	* percent to each duty * fly	sup	afit	pme	attrit
0	0	0.0	0.0	0.0	0.0	0.0
1	1	100.0	0.0	0.0	0.0	0.0
2	2	0.0	0.0	0.0	0.0	0.0
3	3	100.0	0.0	0.0	0.0	0.0
4	4	100.0	0.0	0.0	0.0	0.0
5	5	100.0	0.0	0.0	0.0	0.0
6	6	17.9	68.3	0.0	2.1	11.7
7	6	52.7	10.9	32.7	3.6	0.0
7	7	22.7	27.3	0.0	0.0	50.0
8	6	33.3	66.7	0.0	0.0	0.0
8	7	8.6	62.9	28.6	0.0	0.0
8	8	8.1	27.0	0.0	0.0	64.9
9	6	0.0	100.0	0.0	0.0	0.0
9	7	16.7	83.3	0.0	0.0	0.0
9	8	12.5	68.8	0.0	0.0	18.8
9	9	0.0	9.1	0.0	0.0	90.9
10	6	0.0	90.0	10.0	0.0	0.0
10	7	0.0	71.4	0.0	0.0	28.6
10	8	50.0	50.0	0.0	0.0	0.0
10	9	60.0	20.0	0.0	0.0	20.0
10	10	11.1	0.0	0.0	0.0	88.9
11	6	0.0	83.3	0.0	0.0	16.7
11	7	0.0	66.7	0.0	0.0	33.3
11	8	0.0	70.0	0.0	0.0	30.0
11	9	0.0	85.7	0.0	0.0	14.3
11	10	0.0	66.7	0.0	0.0	33.3
11	11	0.0	76.9	0.0	0.0	23.1
12	6	100.0	0.0	0.0	0.0	0.0
12	7	75.0	0.0	0.0	0.0	25.0
12	8	0.0	0.0	0.0	0.0	0.0
12	9	0.0	91.7	0.0	0.0	8.3
12	10	0.0	80.0	0.0	0.0	20.0
12	11	0.0	72.0	16.0	0.0	12.0
13	6	0.0	0.0	0.0	0.0	0.0
13	7	0.0	0.0	0.0	0.0	0.0
13	8	0.0	100.0	0.0	0.0	0.0
13	9	0.0	100.0	0.0	0.0	0.0
13	10	0.0	100.0	0.0	0.0	0.0
13	11	0.0	97.0	0.0	0.0	3.0
14	6	100.0	0.0	0.0	0.0	0.0
14	7	0.0	0.0	0.0	0.0	0.0
14	8	0.0	75.0	0.0	25.0	0 0
14	9	0.0	100.0	0.0	0.0	0.0
14	10	0.0	100.0	0.0	0.0	0.0
14	11	0.0	54.1	0.0	45.9	0.0
15	6	0.0	0.0	0.0	0.0	0.0

AD-A189 511 A NETWORK APPROACH TO RATED OFFICER GATE MANAGEMENT(U) 3/3
AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL
OF ENGINEERING M S OLSON DEC 87 AFIT/GOR/ENS/87D-13

UNCLASSIFIED F/G 5/9 NL


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5

15	7	0.0	0.0	0.0	0.0	0.0
15	8	100.0	0.0	0.0	0.0	0.0
15	9	100.0	0.0	0.0	0.0	0.0
15	10	100.0	0.0	0.0	0.0	0.0
15	11	78.6	21.4	0.0	0.0	0.0
16	6	0.0	0.0	0.0	0.0	0.0
16	7	0.0	0.0	0.0	0.0	0.0
16	8	100.0	0.0	0.0	0.0	0.0
16	9	100.0	0.0	0.0	0.0	0.0
16	10	100.0	0.0	0.0	0.0	0.0
16	11	44.3	52.5	0.0	0.0	3.3
17	6	0.0	0.0	0.0	0.0	0.0
17	7	0.0	0.0	0.0	0.0	0.0
17	8	0.0	0.0	0.0	0.0	0.0
17	9	100.0	0.0	0.0	0.0	0.0
17	10	100.0	0.0	0.0	0.0	0.0
17	11	22.0	65.9	0.0	9.8	2.4
18	6	100.0	0.0	0.0	0.0	0.0
18	7	0.0	0.0	0.0	0.0	0.0
18	8	100.0	0.0	0.0	0.0	0.0
18	9	100.0	0.0	0.0	0.0	0.0
18	10	100.0	0.0	0.0	0.0	0.0
18	11	33.3	33.5	0.0	0.0	33.3

optimal assignment percentages for fiscal year 89

asd group	fly credit	* percent to each duty * fly	sup	afit	pme	attrit
0	0	0.0	0.0	0.0	0.0	0.0
1	1	100.0	0.0	0.0	0.0	0.0
2	2	100.0	0.0	0.0	0.0	0.0
3	3	100.0	0.0	0.0	0.0	0.0
4	4	100.0	0.0	0.0	0.0	0.0
5	5	100.0	0.0	0.0	0.0	0.0
6	6	58.3	0.0	0.0	0.0	41.7
7	6	100.0	0.0	0.0	0.0	0.0
7	7	18.1	20.5	0.0	6.0	55.4
8	6	5.0	95.0	0.0	0.0	0.0
8	7	23.5	35.3	11.8	0.0	29.4
8	8	0.0	12.0	0.0	0.0	88.0
9	6	0.0	0.0	100.0	0.0	0.0
9	7	6.3	93.8	0.0	0.0	0.0
9	8	16.7	55.6	0.0	0.0	27.8
9	9	0.0	7.7	0.0	0.0	92.3
10	6	100.0	0.0	0.0	0.0	0.0
10	7	0.0	60.0	0.0	0.0	40.0
10	8	0.0	100.0	0.0	0.0	0.0
10	9	50.0	0.0	25.0	0.0	25.0
10	10	0.0	22.2	0.0	0.0	77.8
11	6	0.0	20.0	60.0	0.0	20.0
11	7	0.0	0.0	0.0	0.0	100.0
11	8	0.0	0.0	33.3	0.0	66.7
11	9	0.0	0.0	66.7	0.0	33.3
11	10	0.0	0.0	57.1	0.0	42.9
11	11	0.0	0.0	57.1	0.0	42.9
12	6	0.0	0.0	0.0	0.0	0.0
12	7	0.0	0.0	85.7	0.0	14.3
12	8	50.0	0.0	0.0	0.0	50.0
12	9	0.0	0.0	80.0	0.0	20.0
12	10	66.7	0.0	0.0	0.0	33.3
12	11	0.0	57.1	0.0	0.0	42.9
13	6	0.0	0.0	0.0	0.0	0.0
13	7	0.0	0.0	0.0	100.0	0.0
13	8	0.0	33.3	0.0	66.7	0.0
13	9	0.0	0.0	0.0	100.0	0.0
13	10	0.0	0.0	0.0	100.0	0.0
13	11	0.0	44.4	0.0	44.4	11.1
14	6	0.0	0.0	0.0	0.0	0.0
14	7	100.0	0.0	0.0	0.0	0.0
14	8	33.3	66.7	0.0	0.0	0.0
14	9	100.0	0.0	0.0	0.0	0.0
14	10	100.0	0.0	0.0	0.0	0.0
14	11	35.3	58.8	0.0	0.0	5.9
15	6	0.0	0.0	0.0	0.0	0.0

15	7	0.0	0.0	0.0	0.0	0.0
15	8	100.0	0.0	0.0	0.0	0.0
15	9	100.0	0.0	0.0	0.0	0.0
15	10	100.0	0.0	0.0	0.0	0.0
15	11	0.0	100.0	0.0	0.0	0.0
16	6	0.0	0.0	0.0	0.0	0.0
16	7	0.0	0.0	0.0	0.0	0.0
16	8	0.0	0.0	0.0	0.0	0.0
16	9	100.0	0.0	0.0	0.0	0.0
16	10	100.0	0.0	0.0	0.0	0.0
16	11	0.0	93.9	0.0	0.0	6.1
17	6	0.0	0.0	0.0	0.0	0.0
17	7	0.0	0.0	0.0	0.0	0.0
17	8	0.0	0.0	0.0	0.0	0.0
17	9	100.0	0.0	0.0	0.0	0.0
17	10	100.0	0.0	0.0	0.0	0.0
17	11	37.5	37.5	0.0	25.0	0.0
18	6	0.0	0.0	0.0	0.0	0.0
18	7	100.0	0.0	0.0	0.0	0.0
18	8	0.0	0.0	0.0	0.0	0.0
18	9	100.0	0.0	0.0	0.0	0.0
18	10	100.0	0.0	0.0	0.0	0.0
18	11	10.5	20.2	0.0	0.0	69.3

optimal assignment percentages for fiscal year 90

asd	fly	* percent to each duty *				
group	credit	fly	sup	afit	pme	attrit
0	0	0.0	0.0	0.0	0.0	0.0
1	1	100.0	0.0	0.0	0.0	0.0
2	2	0.0	0.0	0.0	0.0	0.0
3	3	100.0	0.0	0.0	0.0	0.0
4	4	100.0	0.0	0.0	0.0	0.0
5	5	100.0	0.0	0.0	0.0	0.0
6	6	0.0	11.5	40.4	9.6	38.5
7	6	0.0	0.0	0.0	0.0	0.0
7	7	0.0	63.4	0.0	0.0	36.6
8	6	0.0	100.0	0.0	0.0	0.0
8	7	0.0	100.0	0.0	0.0	0.0
8	8	0.0	30.9	0.0	0.0	69.1
9	6	0.0	100.0	0.0	0.0	0.0
9	7	0.0	100.0	0.0	0.0	0.0
9	8	0.0	46.7	0.0	0.0	53.3
9	9	0.0	0.0	0.0	0.0	100.0
10	6	0.0	100.0	0.0	0.0	0.0
10	7	0.0	0.0	0.0	0.0	100.0
10	8	0.0	100.0	0.0	0.0	0.0
10	9	0.0	81.8	0.0	0.0	18.2
10	10	0.0	0.0	0.0	0.0	100.0
11	6	0.0	50.0	0.0	0.0	50.0
11	7	0.0	66.7	0.0	0.0	33.3
11	8	0.0	0.0	33.3	0.0	66.7
11	9	0.0	50.0	25.0	0.0	25.0
11	10	0.0	0.0	50.0	0.0	50.0
11	11	0.0	25.0	0.0	0.0	75.0
12	6	0.0	60.0	40.0	0.0	0.0
12	7	0.0	0.0	0.0	100.0	0.0
12	8	0.0	75.0	25.0	0.0	0.0
12	9	0.0	40.0	40.0	0.0	20.0
12	10	0.0	100.0	0.0	0.0	0.0
12	11	0.0	76.9	0.0	0.0	23.1
13	6	0.0	100.0	0.0	0.0	0.0
13	7	0.0	75.0	0.0	25.0	0.0
13	8	0.0	100.0	0.0	0.0	0.0
13	9	0.0	50.0	0.0	50.0	0.0
13	10	0.0	100.0	0.0	0.0	0.0
13	11	0.0	8.3	0.0	83.3	8.3
14	6	0.0	0.0	0.0	0.0	0.0
14	7	0.0	100.0	0.0	0.0	0.0
14	8	0.0	100.0	0.0	0.0	0.0
14	9	0.0	100.0	0.0	0.0	0.0
14	10	0.0	100.0	0.0	0.0	0.0
14	11	0.0	88.9	0.0	11.1	0.0
15	6	0.0	0.0	0.0	0.0	0.0

15	7	0.0	0.0	0.0	0.0	0.0
15	8	100.0	0.0	0.0	0.0	0.0
15	9	100.0	0.0	0.0	0.0	0.0
15	10	100.0	0.0	0.0	0.0	0.0
15	11	0.0	100.0	0.0	0.0	0.0
16	6	0.0	0.0	0.0	0.0	0.0
16	7	0.0	0.0	0.0	0.0	0.0
16	8	0.0	0.0	0.0	0.0	0.0
16	9	100.0	0.0	0.0	0.0	0.0
16	10	100.0	0.0	0.0	0.0	0.0
16	11	0.0	88.2	0.0	0.0	11.8
17	6	0.0	0.0	0.0	0.0	0.0
17	7	100.0	0.0	0.0	0.0	0.0
17	8	0.0	0.0	0.0	0.0	0.0
17	9	100.0	0.0	0.0	0.0	0.0
17	10	100.0	0.0	0.0	0.0	0.0
17	11	0.0	52.9	0.0	47.1	0.0
18	6	0.0	0.0	0.0	0.0	0.0
18	7	0.0	0.0	0.0	0.0	0.0
18	8	100.0	0.0	0.0	0.0	0.0
18	9	100.0	0.0	0.0	0.0	0.0
18	10	100.0	0.0	0.0	0.0	0.0
18	11	0.0	14.2	0.0	0.3	85.5

optimal assignment percentages for fiscal year 91

asd group	fly credit	* percent to each duty * fly	sup	afit	pme	attrit
0	0	0.0	0.0	0.0	0.0	0.0
1	1	100.0	0.0	0.0	0.0	0.0
2	2	0.0	0.0	0.0	0.0	0.0
3	3	0.0	0.0	0.0	0.0	0.0
4	4	100.0	0.0	0.0	0.0	0.0
5	5	0.0	0.0	0.0	0.0	0.0
6	6	0.0	78.3	0.0	0.0	21.7
7	6	0.0	100.0	0.0	0.0	0.0
7	7	0.0	64.8	0.0	1.8	33.5
8	6	0.0	0.0	0.0	0.0	0.0
8	7	0.0	0.0	0.0	0.0	0.0
8	8	0.0	66.4	0.0	0.0	33.6
9	6	65.7	0.0	34.3	0.0	0.0
9	7	0.0	0.0	0.0	0.0	0.0
9	8	0.0	0.0	0.0	0.0	0.0
9	9	0.0	0.0	0.0	0.0	100.0
10	6	0.0	100.0	0.0	0.0	0.0
10	7	88.9	0.0	0.0	0.0	11.1
10	8	0.0	0.0	0.0	0.0	0.0
10	9	0.0	93.1	0.0	0.0	6.9
10	10	0.0	0.0	0.0	0.0	100.0
11	6	0.0	50.0	0.0	0.0	50.0
11	7	0.0	95.5	0.0	0.0	4.5
11	8	0.0	80.0	0.0	0.0	20.0
11	9	0.0	0.0	0.0	0.0	100.0
11	10	0.0	0.0	0.0	0.0	100.0
11	11	0.0	0.0	0.0	0.0	100.0
12	6	0.0	100.0	0.0	0.0	0.0
12	7	0.0	100.0	0.0	0.0	0.0
12	8	0.0	8.3	0.0	91.7	0.0
12	9	0.0	50.0	0.0	0.0	50.0
12	10	0.0	75.0	0.0	0.0	25.0
12	11	0.0	0.0	0.0	0.0	100.0
13	6	0.0	18.2	0.0	81.8	0.0
13	7	0.0	100.0	0.0	0.0	0.0
13	8	0.0	100.0	0.0	0.0	0.0
13	9	0.0	100.0	0.0	0.0	0.0
13	10	0.0	0.0	0.0	0.0	0.0
13	11	0.0	87.5	0.0	0.0	12.5
14	6	0.0	100.0	0.0	0.0	0.0
14	7	0.0	100.0	0.0	0.0	0.0
14	8	0.0	100.0	0.0	0.0	0.0
14	9	0.0	100.0	0.0	0.0	0.0
14	10	0.0	100.0	0.0	0.0	0.0
14	11	0.0	100.0	0.0	0.0	0.0
15	6	0.0	0.0	0.0	0.0	0.0

15	7	0.0	0.0	0.0	0.0	0.0
15	8	0.0	0.0	0.0	0.0	0.0
15	9	100.0	0.0	0.0	0.0	0.0
15	10	100.0	0.0	0.0	0.0	0.0
15	11	0.0	100.0	0.0	0.0	0.0
16	6	0.0	0.0	0.0	0.0	0.0
16	7	0.0	0.0	0.0	0.0	0.0
16	8	100.0	0.0	0.0	0.0	0.0
16	9	100.0	0.0	0.0	0.0	0.0
16	10	100.0	0.0	0.0	0.0	0.0
16	11	84.4	15.6	0.0	0.0	0.0
17	6	0.0	0.0	0.0	0.0	0.0
17	7	0.0	0.0	0.0	0.0	0.0
17	8	100.0	0.0	0.0	0.0	0.0
17	9	100.0	0.0	0.0	0.0	0.0
17	10	100.0	0.0	0.0	0.0	0.0
17	11	60.0	0.0	0.0	40.0	0.0
18	6	0.0	0.0	0.0	0.0	0.0
18	7	0.0	0.0	0.0	0.0	0.0
18	8	0.0	0.0	0.0	0.0	0.0
18	9	100.0	0.0	0.0	0.0	0.0
18	10	100.0	0.0	0.0	0.0	0.0
18	11	5.9	44.6	0.0	0.0	49.6

optimal assignment percentages for fiscal year 92

asd group	fly credit	fly	sup	afit	pme	attrit
		* percent to each duty *				
0	0	0.0	0.0	0.0	0.0	0.0
1	1	100.0	0.0	0.0	0.0	0.0
2	2	0.0	0.0	0.0	0.0	0.0
3	3	0.0	0.0	0.0	0.0	0.0
4	4	100.0	0.0	0.0	0.0	0.0
5	5	100.0	0.0	0.0	0.0	0.0
6	6	0.0	73.5	0.0	0.0	26.5
7	6	0.0	0.0	0.0	0.0	0.0
7	7	59.7	12.8	0.0	3.4	24.2
8	6	0.0	0.0	0.0	0.0	0.0
8	7	0.0	0.0	100.0	0.0	0.0
8	8	0.0	0.0	0.0	0.0	100.0
9	6	0.0	0.0	0.0	0.0	0.0
9	7	0.0	0.0	0.0	0.0	0.0
9	8	0.0	0.0	0.0	0.0	0.0
9	9	0.0	0.0	0.0	0.0	100.0
10	6	0.0	100.0	0.0	0.0	0.0
10	7	0.0	5.9	76.5	0.0	17.6
10	8	0.0	0.0	0.0	0.0	0.0
10	9	0.0	0.0	0.0	0.0	100.0
10	10	0.0	0.0	0.0	0.0	100.0
11	6	0.0	0.0	94.7	0.0	5.3
11	7	0.0	83.3	0.0	0.0	16.7
11	8	0.0	0.0	0.0	0.0	100.0
11	9	0.0	0.0	0.0	0.0	100.0
11	10	0.0	0.0	0.0	0.0	100.0
11	11	0.0	0.0	0.0	0.0	0.0
12	6	0.0	0.0	0.0	0.0	0.0
12	7	0.0	0.0	0.0	93.3	6.7
12	8	0.0	0.0	0.0	90.0	10.0
12	9	0.0	0.0	0.0	0.0	100.0
12	10	0.0	0.0	0.0	0.0	100.0
12	11	0.0	0.0	0.0	0.0	100.0
13	6	0.0	0.0	0.0	0.0	0.0
13	7	0.0	100.0	0.0	0.0	0.0
13	8	0.0	100.0	0.0	0.0	0.0
13	9	0.0	100.0	0.0	0.0	0.0
13	10	0.0	100.0	0.0	0.0	0.0
13	11	0.0	75.0	0.0	0.0	25.0
14	6	0.0	100.0	0.0	0.0	0.0
14	7	0.0	0.0	0.0	0.0	0.0
14	8	0.0	0.0	0.0	0.0	0.0
14	9	0.0	0.0	0.0	0.0	0.0
14	10	0.0	0.0	0.0	0.0	0.0
14	11	0.0	0.0	0.0	0.0	0.0
15	6	0.0	0.0	0.0	0.0	0.0

15	7	0.0	0.0	0.0	0.0	0.0
15	8	0.0	0.0	0.0	0.0	0.0
15	9	0.0	0.0	0.0	0.0	0.0
15	10	0.0	0.0	0.0	0.0	0.0
15	11	0.0	100.0	0.0	0.0	0.0
16	6	0.0	0.0	0.0	0.0	0.0
16	7	0.0	0.0	0.0	0.0	0.0
16	8	100.0	0.0	0.0	0.0	0.0
16	9	0.0	0.0	0.0	0.0	0.0
16	10	0.0	0.0	0.0	0.0	0.0
16	11	0.0	100.0	0.0	0.0	0.0
17	6	0.0	0.0	0.0	0.0	0.0
17	7	0.0	0.0	0.0	0.0	0.0
17	8	100.0	0.0	0.0	0.0	0.0
17	9	0.0	0.0	0.0	0.0	0.0
17	10	100.0	0.0	0.0	0.0	0.0
17	11	0.0	100.0	0.0	0.0	0.0
18	6	0.0	0.0	0.0	0.0	0.0
18	7	0.0	0.0	0.0	0.0	0.0
18	8	0.0	0.0	0.0	0.0	0.0
18	9	0.0	0.0	0.0	0.0	0.0
18	10	100.0	0.0	0.0	0.0	0.0
18	11	0.0	56.7	0.0	3.0	40.2

optimal assignment percentages averaged over 88 thru 92

asd group	fly credit	* percent to each duty * fly	sup	afit	pme	attrit
0	0	0.0	0.0	0.0	0.0	0.0
1	1	100.0	0.0	0.0	0.0	0.0
2	2	20.0	0.0	0.0	0.0	0.0
3	3	60.0	0.0	0.0	0.0	0.0
4	4	100.0	0.0	0.0	0.0	0.0
5	5	80.0	0.0	0.0	0.0	0.0
6	6	15.3	46.3	8.1	2.3	28.0
7	6	30.5	22.2	6.5	0.7	0.0
7	7	20.1	37.7	0.0	2.2	39.9
8	6	7.7	52.3	0.0	0.0	0.0
8	7	6.4	39.6	28.1	0.0	5.9
8	8	1.6	27.3	0.0	0.0	71.1
9	6	13.1	40.0	26.9	0.0	0.0
9	7	4.6	55.4	0.0	0.0	0.0
9	8	5.8	34.2	0.0	0.0	20.0
9	9	0.0	3.4	0.0	0.0	96.6
10	6	20.0	78.0	2.0	0.0	0.0
10	7	17.8	27.5	15.3	0.0	39.5
10	8	10.0	50.0	0.0	0.0	0.0
10	9	22.0	39.0	5.0	0.0	34.0
10	10	2.2	4.4	0.0	0.0	93.3
11	6	0.0	40.7	30.9	0.0	28.4
11	7	0.0	62.4	0.0	0.0	37.6
11	8	0.0	30.0	13.3	0.0	56.7
11	9	0.0	27.1	18.3	0.0	54.5
11	10	0.0	13.3	21.4	0.0	65.2
11	11	0.0	20.4	11.4	0.0	48.2
12	6	20.0	32.0	8.0	0.0	0.0
12	7	15.0	20.0	17.1	38.7	9.2
12	8	10.0	16.7	5.0	36.3	12.0
12	9	0.0	36.3	24.0	0.0	39.7
12	10	13.3	51.0	0.0	0.0	35.7
12	11	0.0	41.2	3.2	0.0	55.6
13	6	0.0	23.6	0.0	16.4	0.0
13	7	0.0	55.0	0.0	25.0	0.0
13	8	0.0	86.7	0.0	13.3	0.0
13	9	0.0	70.0	0.0	30.0	0.0
13	10	0.0	60.0	0.0	20.0	0.0
13	11	0.0	62.4	0.0	25.6	12.0
14	6	20.0	40.0	0.0	0.0	0.0
14	7	20.0	40.0	0.0	0.0	0.0
14	8	6.7	68.3	0.0	5.0	0.0
14	9	20.0	60.0	0.0	0.0	0.0
14	10	20.0	60.0	0.0	0.0	0.0
14	11	7.1	60.4	0.0	11.4	1.2
15	6	0.0	0.0	0.0	0.0	0.0

15	7	0.0	0.0	0.0	0.0	0.0
15	8	60.0	0.0	0.0	0.0	0.0
15	9	80.0	0.0	0.0	0.0	0.0
15	10	80.0	0.0	0.0	0.0	0.0
15	11	15.7	84.3	0.0	0.0	0.0
16	6	0.0	0.0	0.0	0.0	0.0
16	7	0.0	0.0	0.0	0.0	0.0
16	8	60.0	0.0	0.0	0.0	0.0
16	9	80.0	0.0	0.0	0.0	0.0
16	10	80.0	0.0	0.0	0.0	0.0
16	11	25.7	70.1	0.0	0.0	4.2
17	6	0.0	0.0	0.0	0.0	0.0
17	7	20.0	0.0	0.0	0.0	0.0
17	8	40.0	0.0	0.0	0.0	0.0
17	9	80.0	0.0	0.0	0.0	0.0
17	10	100.0	0.0	0.0	0.0	0.0
17	11	23.9	51.3	0.0	24.4	0.5
18	6	20.0	0.0	0.0	0.0	0.0
18	7	20.0	0.0	0.0	0.0	0.0
18	8	40.0	0.0	0.0	0.0	0.0
18	9	80.0	0.0	0.0	0.0	0.0
18	10	100.0	0.0	0.0	0.0	0.0
18	11	9.9	33.8	0.0	0.7	55.6

ASSIGNMENTS THAT RESULT IN MISSED GATES

YEAR	NEW DUTY	ASD	GATE TIME	NUMBER ASSIGNED	UNIT COST	DUTY ROTATING FROM: FLY	SUP	AFIT	PME
1	1	18	6	40.0	2.	40.	0.	0.	0.
1	1	18	8	1.0	1.	0.	1.	0.	0.
1	1	18	9	29.0	1.	2.	27.	0.	0.
2	1	18	7	103.0	2.	103.	0.	0.	0.
2	1	18	9	15.0	1.	1.	14.	0.	0.
3	1	17	7	1.0	1.	0.	1.	0.	0.
3	1	18	8	152.0	1.	152.	0.	0.	0.
3	1	18	9	13.0	1.	1.	12.	0.	0.
4	1	18	9	152.0	1.	152.	0.	0.	0.

OPTIMAL ASSIGNMENT POLICY

broken out by gate attained (hit) or missed
(hit or miss status is as of the start of the assignment)

GATE HIT OR MISSED:			hit	miss	hit	miss	hit	miss
year	duty	none	1	1	2	2	3	3
1	1	677.	89.	0.	84.	41.	227.	54.
1	2	0.	218.	0.	62.	0.	325.	0.
1	3	0.	29.	0.	0.	0.	4.	0.
1	4	0.	6.	0.	0.	0.	25.	0.
2	1	452.	59.	0.	50.	103.	68.	55.
2	2	0.	80.	0.	3.	0.	203.	0.
2	3	0.	17.	0.	12.	0.	4.	0.
2	4	0.	8.	0.	11.	0.	12.	0.
3	1	447.	3.	0.	20.	152.	0.	43.
3	2	0.	147.	0.	34.	0.	111.	0.
3	3	0.	25.	0.	6.	0.	0.	0.
3	4	0.	10.	0.	4.	0.	20.	0.
4	1	528.	90.	0.	52.	0.	59.	165.
4	2	0.	403.	0.	51.	0.	203.	0.
4	3	0.	34.	0.	0.	0.	0.	0.
4	4	0.	25.	0.	0.	0.	8.	0.
5	1	548.	92.	0.	1.	0.	0.	152.
5	2	0.	122.	0.	7.	0.	230.	0.
5	3	0.	36.	0.	0.	0.	0.	0.
5	4	0.	28.	0.	0.	0.	10.	0.

NOTE: duty 1 = fly, duty 2 = staff/supplement,
duty 3 = AFIT, duty 4 = PME/ASTRA

Appendix G: GATES Output File NODEARC.OUT

Note: This is only a portion of the output file that lists all network arcs with the associated beginning and ending nodes.

begin node	time	duty	asd	fly	end node	time	duty	asd	fly	arc num	cost	bound
1	1	1	0	0	586	4	1	3	3	1	0	-1.0
2	1	1	1	1	587	4	1	4	4	2	0	-1.0
3	1	1	2	2	588	4	1	5	5	3	0	-1.0
4	1	1	3	3	589	4	1	6	6	4	0	-1.0
5	1	1	4	4	591	4	1	7	7	5	0	-1.0
6	1	1	5	5	594	4	1	8	8	6	0	-1.0
7	1	1	6	6	598	4	1	9	9	7	0	-1.0
8	1	1	7	6	602	4	1	10	9	8	0	-1.0
9	1	1	7	7	603	4	1	10	10	9	0	-1.0
.					.					.		
.					.					.		
.					.					.		
.					.					.		
67	1	1	18	9	262	2	1	18	10	67	1	-1.0
68	1	1	18	10	263	2	1	18	11	68	0	-1.0
69	1	1	18	11	263	2	1	18	11	69	0	-1.0
70	1	2	6	6	598	4	1	9	9	70	0	-1.0
71	1	2	7	6	602	4	1	10	9	71	0	-1.0
72	1	2	7	7	603	4	1	10	10	72	0	-1.0
.					.					.		
.					.					.		
.					.					.		
.					.					.		
800	5	1	11	8	971	8	1	14	11	668	0	-1.0
801	5	1	11	9	971	8	1	14	11	669	0	-1.0
802	5	1	11	10	971	8	1	14	11	670	0	-1.0
803	5	1	11	11	971	8	1	14	11	671	0	-1.0
905	5	2	18	8	974	6	4	18	8	2367	2	-1.0
906	5	2	18	9	974	6	4	18	9	2368	1	-1.0
907	5	2	18	10	974	6	4	18	10	2369	1	-1.0
908	5	2	18	11	974	6	4	18	11	2370	0	-1.0
971	98	1	0	0	975	99	0	0	0	2371	0	-1.0
972	98	2	0	0	975	99	0	0	0	2372	0	-1.0
973	98	3	0	0	975	99	0	0	0	2373	0	-1.0
974	98	4	0	0	975	99	0	0	0	2374	0	-1.0

Appendix H: Sample of GATES Screen Output

```
$ run gates.exe

         year    lower bound       available         upper bound
         ----    -----------       ---------         -----------
            1    1308.000          1841.000          1948.000
            2    2132.000          2520.000          2771.000
            3    2975.000          3174.000          3625.000
            4    2918.000          3100.000          3556.000
            5    3119.000          3153.000          3799.000

sanity checks passed....building datafiles
total nodes =           975
total flows into network =    4954.000
total attrition out of network =     1801.000
end-of-network sink =   -3153.000
max arc number for end duty    1 is      805
max arc number for end duty    2 is     1745
max arc number for end duty    3 is     2040
max arc number for end duty    4 is     2370
total arcs =          2374

data files built...performing network optimization routine
ITER, OBJ    100   0.32728000E+06
ITER, OBJ    200   0.27468000E+06
ITER, OBJ    300   0.23540000E+06
ITER, OBJ    400   0.12938000E+06
ITER, OBJ    500   0.12150000E+06
ITER, OBJ    600   0.10938800E+06
ITER, OBJ    700   0.87382000E+05
ITER, OBJ    800   0.82122000E+05
ITER, OBJ    900   0.57841667E+05
ITER, OBJ   1000   0.40866000E+05
ITER, OBJ   1100   0.37152833E+05
ITER, OBJ   1200   0.31322333E+05
ITER, OBJ   1300   0.12167000E+05
ITER, OBJ   1400   0.63650000E+04
ITER, OBJ   1500   0.34650000E+04
ITER, OBJ   1600   0.19860000E+04
ITER, OBJ   1700   0.87657143E+03
ITER, OBJ   1800   0.80320000E+03
ITER, OBJ   1900   0.65300000E+03

optimization complete...building output tables
processing complete
a list of nodes and arcs is in file NODEARC.OUT
optimal assignment tables are in file ROTEPLAN.OUT
miscellaneous optimization information is in file FOR007.dat
```

Appendix I: Input/Output Summary

This appendix contains listings of input parameters that were varied as part of the sensitivity analysis of the GATES model. Listed after the input parameters for each model run are the key performance measures and results. Each page in this appendix contains data for a single model run.

For each run, the following input parameters are listed:

1. The run identification number and a brief description;

2. The duty assignment durations used for each duty type;

3. The amount of overmanning (tolerance +) and undermanning (tolerance -) allowed for each duty type, as a percentage of the baseline manning requirements taken from the Rated Management Document (9);

4. The cost associated with missing each flying gate ("gate 0" refers to ASD year groups less than six, "gate 1" is the six-year gate, "gate 2" is the nine-year gate, "gate 3" is the eleven-year gate);

5. Whether a side constraint specifying a manning experience requirement was applied (the only experience requirement applied in any of the runs was the requirement for 50 percent or more of flying duties to be filled by individuals from ASD year group six or higher);

6. Attrition adjustments from the baseline estimates; these adjustments served to shift attrition within a particular ASD year group or to actually reduce attrition for the ASD year group; such adjustments were performed in order to obtain a feasible solution.

The following performance measures and results are listed for each run:

1. Whether the "sanity checks" were passed (these checks are designed to verify whether there is sufficient supply to meet the manning requirements for each year);

2. The number of iterations required by NETSID to obtain the final solution;

3. The final (optimal) objective function value;

4. Whether the problem had a feasible solution;

5. The number of artificial variables, total artificial flow, and cost due to these artificial flows (if the solution was infeasible);

6. The difference between the objective function value and the artificial costs;

7. Rotations into the network for each year;

8. Gains (UFT and FAIP) into the network for each year;

9. Total attrition for each year;

10. The number of individuals that have already missed their gates at time of initial rotation into the network;

11. The total number of assignments made (flows) each year, broken out by duty type;

12. The solution flows (assignments) which incur a cost due to failure to meet a gate by the end of the tour of duty (these are broken out by gate missed, year, and flying/nonflying duties).

The run identifying numbers are categorized according to the tour lengths for flying duties and staff/supplement duties as follows:

run numbers	flying tour	staff/supplement tour
A1 thru A14	3 years	3 years
B1 thru B2	4 years	4 years

In all cases, one year was used for the tour length for AFIT and PME. Here is a brief summary of the key features of each run:

run	general description
A1	baseline parameters, infeasible
A2	adjusted attrition, feasible solution
A3	double value of all missed gate costs
A4	equal costs for gates 0 and 1
A5	equal costs for gates 0, 1, and 2

A6	equal costs for all gates
A7	equal costs for gates 0 and 1, equal costs for gates 2 and 3
A8	zero cost for gate 0
A9	equal costs for gates 0 and 1, higher cost for gate 2
A10	same as A2 except no experience requirement for flying duty manning
A11	reduced undermanning tolerance for flying duties, failed "sanity" checks
A12	changed tolerance to pass checks
A13	reduced undermanning tolerance for staff/supplement duties, failed "sanity" checks
A14	changed tolerance to pass checks
B1	same as A1 except duty durations, infeasible
B2	adjusted attrition, still infeasible

RUN #	A1 (70)
DESCRIPTION:	baseline parameters, infeasible

DUTY TYPE:	FLY	SUP	AFIT	PME	
Duration (years)	3	3	1	1	
Tolerance + (%)	10%	10%	5%	0%	
Tolerance - (%)	10%	10%	5%	0%	
GATE:	Gate 0	Gate 1	Gate 2	Gate 3	
Gate Miss Cost	4	3	2	1	
EXPERIENCE RQMT	Yes				
ATTRITION ADJUSTMENTS	"Baseline": attrition distribution based on examination of input data file.				

SANITY CHECKS	Pass
# ITERATIONS	1868
OBJ FUNCTION VALUE	6267
FEASIBLE ?	No
ARTIFICIALS: Number	29 variables
Total artif flow	154.0
Cost of artificials	6160
OBJ $ - ARTIFICIAL $	107

YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	1795	831	783	0	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	277	465	443	327	293
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	N/A	N/A
Missed 2nd Gate	41	63	49	N/A	N/A
Missed 3rd Gate	54	25	28	N/A	N/A
TOTAL ASSIGNMENTS	1797	1062	930	1570	1146
To FLY duties	1172	716	496	985	731
To SUP duties	561	283	369	522	337
To AFIT duties	33	32	31	30	40
To PME duties	31	31	34	33	38
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	0	0	0	0	0
Will Miss GATE 1	0	0	0	0	0
Will Miss GATE 2	0	0	0	0	0
Will Miss GATE 3	15/37	0/32	14/1	8/0	0

RUN #	A2 (72)				
DESCRIPTION:	adjusted attrition, feasible				
DUTY TYPE:	FLY	SUP	AFIT	PME	
Duration (years)	3	3	1	1	
Tolerance + (%)	10%	10%	5%	0%	
Tolerance - (%)	10%	10%	5%	0%	
GATE:	Gate 0	Gate 1	Gate 2	Gate 3	
Gate Miss Cost	4	3	2	1	
EXPERIENCE RQMT	Yes				
ATTRITION ADJUSTMENTS	As required to achieve feasible solution. Adjustments were within ASD groups (usually), but sometimes shifted to another year.				
SANITY CHECKS	Pass				
# ITERATIONS	1917				
OBJ FUNCTION VALUE	649				
FEASIBLE ?	Yes				
ARTIFICIALS: Number	N/A				
Total artif flow	N/A				
Cost of artificials	N/A				
OBJ $ - ARTIFICIAL $	649				
YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	1795	831	783	0	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	271	467	444	373	246
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	N/A	N/A
Missed 2nd Gate	41	63	49	N/A	N/A
Missed 3rd Gate	54	25	28	N/A	N/A
TOTAL ASSIGNMENTS	1841	1137	1022	1618	1226
To FLY duties	1172	787	665	894	793
To SUP duties	605	286	292	657	359
To AFIT duties	33	33	31	34	36
To PME duties	31	31	34	33	38
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	0	0	0	0	0
Will Miss GATE 1	0	0	0	0	0
Will Miss GATE 2	40/0	103/0	0	0	0
Will Miss GATE 3	30/0	15/0	166/0	152/0	0

RUN #	A3 (73)
DESCRIPTION:	double values of all gate costs

DUTY TYPE:	FLY	SUP	AFIT	PME
Duration (years)	3	3	1	1
Tolerance + (%)	10%	10%	5%	0%
Tolerance - (%)	10%	10%	5%	0%

GATE:	Gate 0	Gate 1	Gate 2	Gate 3
Gate Miss Cost	8	6	4	2

EXPERIENCE RQMT	Yes
ATTRITION ADJUSTMENTS	Same as run # A2

SANITY CHECKS	Pass
# ITERATIONS	2046
OBJ FUNCTION VALUE	1298 (exactly double run # A2)
FEASIBLE ?	Yes
ARTIFICIALS: Number	N/A
Total artif flow	N/A
Cost of artificials	N/A
OBJ $ - ARTIFICIAL $	1298

YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	1795	831	783	0	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	271	467	444	373	246
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	N/A	N/A
Missed 2nd Gate	41	63	49	N/A	N/A
Missed 3rd Gate	54	25	28	N/A	N/A
TOTAL ASSIGNMENTS	1841	1137	1022	1618	1226
To FLY duties	1172	787	665	894	793
To SUP duties	605	286	292	657	359
To AFIT duties	33	33	31	34	36
To PME duties	31	31	34	33	38
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	0	0	0	0	0
Will Miss GATE 1	0	0	0	0	0
Will Miss GATE 2	40/0	103/0	0	0	0
Will Miss GATE 3	30/0	15/0	166/0	152/0	0

RUN #	A4 (74)				
DESCRIPTION:	equal costs for gates 0 and 1				
DUTY TYPE:	FLY	SUP	AFIT	PME	
Duration (years)	3	3	1	1	
Tolerance + (%)	10%	10%	5%	0%	
Tolerance - (%)	10%	10%	5%	0%	
GATE:	Gate 0	Gate 1	Gate 2	Gate 3	
Gate Miss Cost	3	3	2	1	
EXPERIENCE RQMT	Yes				
ATTRITION ADJUSTMENTS	Same as run # A2				
SANITY CHECKS	Pass				
# ITERATIONS	1917				
OBJ FUNCTION VALUE	649				
FEASIBLE ?	Yes				
ARTIFICIALS: Number	N/A				
Total artif flow	N/A				
Cost of artificials	N/A				
OBJ $ - ARTIFICIAL $	649				
YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	1795	831	783	0	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	271	467	444	373	246
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	N/A	N/A
Missed 2nd Gate	41	63	49	N/A	N/A
Missed 3rd Gate	54	25	28	N/A	N/A
TOTAL ASSIGNMENTS	1841	1137	1022	1618	1226
To FLY duties	1172	787	665	894	793
To SUP duties	605	286	292	657	359
To AFIT duties	33	33	31	34	36
To PME duties	31	31	34	33	38
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	0	0	0	0	0
Will Miss GATE 1	0	0	0	0	0
Will Miss GATE 2	40/0	103/0	0	0	0
Will Miss GATE 3	30/0	15/0	166/0	152/0	0

RUN #	A5 (75)				
DESCRIPTION:	equal costs for gates 0, 1 ,and 2				
DUTY TYPE:	FLY	SUP	AFIT	PME	
Duration (years)	3	3	1	1	
Tolerance + (%)	10%	10%	5%	0%	
Tolerance - (%)	10%	10%	5%	0%	
GATE:	Gate 0	Gate 1	Gate 2	Gate 3	
Gate Miss Cost	3	3	3	1	
EXPERIENCE RQMT	Yes				
ATTRITION ADJUSTMENTS	Same as run # A2				
SANITY CHECKS	Pass				
# ITERATIONS	1899				
OBJ FUNCTION VALUE	792				
FEASIBLE ?	Yes				
ARTIFICIALS: Number	N/A				
Total artif flow	N/A				
Cost of artificials	N/A				
OBJ $ - ARTIFICIAL $	792				
YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	1795	831	783	0	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	271	467	444	373	246
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	N/A	N/A
Missed 2nd Gate	41	63	49	N/A	N/A
Missed 3rd Gate	54	25	28	N/A	N/A
TOTAL ASSIGNMENTS	1841	1137	1022	1610	1218
To FLY duties	1161	825	665	855	848
To SUP duties	616	248	292	688	296
To AFIT duties	33	33	31	34	36
To PME duties	31	31	34	33	38
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	0	0	0	0	0
Will Miss GATE 1	0	0	0	0	0
Will Miss GATE 2	40/0	103/0	0	0	0
Will Miss GATE 3	30/0	15/0	166/0	152/0	0

RUN #	A6 (76)				
DESCRIPTION:	equal costs for all gates				
DUTY TYPE:	FLY	SUP	AFIT	PME	
Duration (years)	3	3	1	1	
Tolerance + (%)	10%	10%	5%	0%	
Tolerance - (%)	10%	10%	5%	0%	
GATE:	Gate 0	Gate 1	Gate 2	Gate 3	
Gate Miss Cost	3	3	3	3	
EXPERIENCE RQMT	Yes				
ATTRITION ADJUSTMENTS	Same as run # A2				
SANITY CHECKS	Pass				
# ITERATIONS	2032				
OBJ FUNCTION VALUE	1518				
FEASIBLE ?	Yes				
ARTIFICIALS: Number	N/A				
Total artif flow	N/A				
Cost of artificials	N/A				
OBJ $ - ARTIFICIAL $	1518				
YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	1795	831	783	0	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	271	467	444	373	246
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	N/A	N/A
Missed 2nd Gate	41	63	49	N/A	N/A
Missed 3rd Gate	54	25	28	N/A	N/A
TOTAL ASSIGNMENTS	1841	1137	1022	1610	1218
To FLY duties	1172	815	664	884	890.5
To SUP duties	605	258	293	659	249.5
To AFIT duties	33	33	31	34	40
To PME duties	31	31	34	33	38
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	0	0	0	0	0
Will Miss GATE 1	0	0	0	0	0
Will Miss GATE 2	40/0	103/0	0/1	0	0
Will Miss GATE 3	30/0	15/0	165/0	152/0	0

RUN #	A7 (77)				
DESCRIPTION:	equal costs gates 0 & 1, gates 2 & 3				
DUTY TYPE:	FLY	SUP	AFIT	PME	
Duration (years)	3	3	1	1	
Tolerance + (%)	10%	10%	5%	0%	
Tolerance - (%)	10%	10%	5%	0%	
GATE:	Gate 0	Gate 1	Gate 2	Gate 3	
Gate Miss Cost	3	3	2	2	
EXPERIENCE RQMT	Yes				
ATTRITION ADJUSTMENTS	Same as run # A2				
SANITY CHECKS	Pass				
# ITERATIONS	1751				
OBJ FUNCTION VALUE	1012				
FEASIBLE ?	Yes				
ARTIFICIALS: Number	N/A				
Total artif flow	N/A				
Cost of artificials	N/A				
OBJ $ - ARTIFICIAL $	1012				
YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	1795	831	783	0	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	271	467	444	373	246
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	N/A	N/A
Missed 2nd Gate	41	63	49	N/A	N/A
Missed 3rd Gate	54	25	28	N/A	N/A
TOTAL ASSIGNMENTS	1841	1137	1022	1618	1214
To FLY duties	1162	825	664	989	877
To SUP duties	615	248	293	566	263
To AFIT duties	33	33	31	30	36
To PME duties	31	31	34	33	38
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	0	0	0	0	0
Will Miss GATE 1	0	0	0	0	0
Will Miss GATE 2	40/0	103/0	0/1	0	0
Will Miss GATE 3	30/0	15/0	165/0	152/0	0

RUN #	A8 (78)				
DESCRIPTION:	zero cost for gate 0				

DUTY TYPE:	FLY	SUP	AFIT	PME
Duration (years)	3	3	1	1
Tolerance + (%)	10%	10%	5%	0%
Tolerance - (%)	10%	10%	5%	0%

GATE:	Gate 0	Gate 1	Gate 2	Gate 3
Gate Miss Cost	0	3	2	1

EXPERIENCE RQMT	Yes
ATTRITION ADJUSTMENTS	Same as run # A2

SANITY CHECKS	Pass
# ITERATIONS	1917
OBJ FUNCTION VALUE	649
FEASIBLE ?	Yes
ARTIFICIALS: Number	N/A
Total artif flow	N/A
Cost of artificials	N/A
OBJ $ - ARTIFICIAL $	649

YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	1795	831	783	0	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	271	467	444	373	246
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	N/A	N/A
Missed 2nd Gate	41	63	49	N/A	N/A
Missed 3rd Gate	54	25	28	N/A	N/A
TOTAL ASSIGNMENTS	1841	1137	1022	1618	1226
To FLY duties	1172	787	665	894	793
To SUP duties	605	286	292	657	359
To AFIT duties	33	33	31	34	36
To PME duties	31	31	34	33	38
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	0	0	0	0	0
Will Miss GATE 1	0	0	0	0	0
Will Miss GATE 2	40/0	103/0	0	0	0
Will Miss GATE 3	30/0	15/0	166/0	152/0	0

RUN #	A9 (79)				
DESCRIPTION:	equal costs gates 0 & 1, higher #2				
DUTY TYPE:	FLY	SUP	AFIT	PME	
Duration (years)	3	3	1	1	
Tolerance + (%)	10%	10%	5%	0%	
Tolerance - (%)	10%	10%	5%	0%	
GATE:	Gate 0	Gate 1	Gate 2	Gate 3	
Gate Miss Cost	3	3	4	1	
EXPERIENCE RQMT	Yes				
ATTRITION ADJUSTMENTS	Same as run # A2				
SANITY CHECKS	Pass				
# ITERATIONS	1845				
OBJ FUNCTION VALUE	935				
FEASIBLE ?	Yes				
ARTIFICIALS: Number	N/A				
Total artif flow	N/A				
Cost of artificials	N/A				
OBJ $ - ARTIFICIAL $	935				
YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	1795	831	783	0	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	271	467	444	373	246
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	N/A	N/A
Missed 2nd Gate	41	63	49	N/A	N/A
Missed 3rd Gate	54	25	28	N/A	N/A
TOTAL ASSIGNMENTS	1841	1137	1022	1610	1218
To FLY duties	1165	829	665	887	835
To SUP duties	612	244	292	656	309
To AFIT duties	33	33	31	34	36
To PME duties	31	31	34	33	38
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	0	0	0	0	0
Will Miss GATE 1	0	0	0	0	0
Will Miss GATE 2	40/0	103/0	0	0	0
Will Miss GATE 3	30/0	15/0	166/0	152/0	0

RUN #	A10 (80)
DESCRIPTION:	no experience requirement

DUTY TYPE:	FLY	SUP	AFIT	PME
Duration (years)	3	3	1	1
Tolerance + (%)	10%	10%	5%	0%
Tolerance - (%)	10%	10%	5%	0%

GATE:	Gate 0	Gate 1	Gate 2	Gate 3
Gate Miss Cost	4	3	2	1

EXPERIENCE RQMT	No
ATTRITION ADJUSTMENTS	Same as run # A2

SANITY CHECKS	Pass
# ITERATIONS	1891
OBJ FUNCTION VALUE	649
FEASIBLE ?	Yes
ARTIFICIALS: Number	N/A
Total artif flow	N/A
Cost of artificials	N/A
OBJ $ - ARTIFICIAL $	649

YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	1795	831	783	0	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	271	467	444	373	246
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	N/A	N/A
Missed 2nd Gate	41	63	49	N/A	N/A
Missed 3rd Gate	54	25	28	N/A	N/A
TOTAL ASSIGNMENTS	1841	1137	1020	1608	1210
To FLY duties	1172	774	665	980	763
To SUP duties	605	299	290	561	373
To AFIT duties	33	33	31	34	36
To PME duties	31	31	34	33	38
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	0	0	0	0	0
Will Miss GATE 1	0	0	0	0	0
Will Miss GATE 2	40/0	103/0	0	0	0
Will Miss GATE 3	30/0	15/0	166/0	152/0	0

RUN #	A11 (81)
DESCRIPTION:	lower underman tolerance (flying)

DUTY TYPE:	FLY	SUP	AFIT	PME
Duration (years)	3	3	1	1
Tolerance + (%)	10%	10%	5%	0%
Tolerance - (%)	5%	10%	5%	0%

GATE:	Gate 0	Gate 1	Gate 2	Gate 3
Gate Miss Cost	4	3	2	1

EXPERIENCE RQMT	Yes
ATTRITION ADJUSTMENTS	Same as run # A2

SANITY CHECKS	FAIL, insufficient personnel, year 5
# ITERATIONS	N/A
OBJ FUNCTION VALUE	N/A
FEASIBLE ?	N/A
ARTIFICIALS: Number	N/A
Total artif flow	N/A
Cost of artificials	N/A
OBJ $ - ARTIFICIAL $	N/A

YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	1795	831	783	0	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	271	467	444	373	246
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	N/A	N/A
Missed 2nd Gate	41	63	49	N/A	N/A
Missed 3rd Gate	54	25	28	N/A	N/A

TOTAL ASSIGNMENTS	N/A
To FLY duties	
To SUP duties	
To AFIT duties	
To PME duties	N/A

ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)
Will Miss GATE 0	N/A
Will Miss GATE 1	
Will Miss GATE 2	
Will Miss GATE 3	N/A

RUN #	A12 (82)				
DESCRIPTION:	changed tolerance to pass checks				
DUTY TYPE:	FLY	SUP	AFIT	PME	
Duration (years)	3	3	1	1	
Tolerance + (%)	10%	10%	5%	0%	
Tolerance - (%)	5%*	10%	5%	0%	
	* Tolerance -10% (FLY) in year 5				
GATE:	Gate 0	Gate 1	Gate 2	Gate 3	
Gate Miss Cost	4	3	2	1	
EXPERIENCE RQMT	Yes				
ATTRITION ADJUSTMENTS	Same as run # A2				
SANITY CHECKS	Pass				
# ITERATIONS	1900				
OBJ FUNCTION VALUE	649				
FEASIBLE ?	Yes				
ARTIFICIALS: Number	N/A				
Total artif flow	N/A				
Cost of artificials	N/A				
OBJ $ - ARTIFICIAL $	649				
YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	1795	831	783	0	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	271	467	444	373	246
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	N/A	N/A
Missed 2nd Gate	41	63	49	N/A	N/A
Missed 3rd Gate	54	25	28	N/A	N/A
TOTAL ASSIGNMENTS	1841	1129	1014	1610	1218
To FLY duties	1172	759	686	1103	747
To SUP duties	605	306	263	440	397
To AFIT duties	33	33	31	34	36
To PME duties	31	31	34	33	38
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	0	0	0	0	0
Will Miss GATE 1	0	0	0	0	0
Will Miss GATE 2	40/0	103/0	0	0	0
Will Miss GATE 3	30/0	15/0	166/0	152/0	0

RUN #	A13 (83)
DESCRIPTION:	lower underman tolerance (staff)

DUTY TYPE:	FLY	SUP	AFIT	PME
Duration (years)	3	3	1	1
Tolerance + (%)	10%	10%	5%	0%
Tolerance - (%)	10%	5%	5%	0%

GATE:	Gate 0	Gate 1	Gate 2	Gate 3
Gate Miss Cost	4	3	2	1

EXPERIENCE RQMT	Yes
ATTRITION ADJUSTMENTS	Same as run # A2

SANITY CHECKS	FAIL
# ITERATIONS	N/A
OBJ FUNCTION VALUE	N/A
FEASIBLE ?	N/A
ARTIFICIALS: Number	N/A
Total artif flow	N/A
Cost of artificials	N/A
OBJ $ - ARTIFICIAL $	N/A

YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	1795	831	783	0	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	271	467	444	373	246
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	N/A	N/A
Missed 2nd Gate	41	63	49	N/A	N/A
Missed 3rd Gate	54	25	28	N/A	N/A
TOTAL ASSIGNMENTS	N/A				
To FLY duties					
To SUP duties					
To AFIT duties					
To PME duties					N/A
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	N/A				
Will Miss GATE 1					
Will Miss GATE 2					
Will Miss GATE 3					N/A

RUN #	A14 (84)
DESCRIPTION:	changed tolerance to pass checks

DUTY TYPE:	FLY	SUP	AFIT	PME	
Duration (years)	3	3	1	1	
Tolerance + (%)	10%	10%	5%	0%	
Tolerance - (%)	10%	5%*	5%	0%	
	* Tolerance -10% (SUP) in year 5				
GATE:	Gate 0	Gate 1	Gate 2	Gate 3	
Gate Miss Cost	4	3	2	1	
EXPERIENCE RQMT	Yes				
ATTRITION ADJUSTMENTS	Same as run # A2				

SANITY CHECKS	Pass				
# ITERATIONS	1777				
OBJ FUNCTION VALUE	649				
FEASIBLE ?	Yes				
ARTIFICIALS: Number	N/A				
Total artif flow	N/A				
Cost of artificials	N/A				
OBJ $ - ARTIFICIAL $	649				
YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	1795	831	783	0	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	271	467	444	373	246
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	N/A	N/A
Missed 2nd Gate	41	63	49	N/A	N/A
Missed 3rd Gate	54	25	28	N/A	N/A
TOTAL ASSIGNMENTS	1841	1137	1022	1610	1218
To FLY duties	1065	813	665	900	888
To SUP duties	712	260	292	643	256
To AFIT duties	33	33	31	34	36
To PME duties	31	31	34	33	38
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	0	0	0	0	0
Will Miss GATE 1	0	0	0	0	0
Will Miss GATE 2	40/0	103/0	0	0	0
Will Miss GATE 3	30/0	15/0	166/0	152/0	0

RUN #	B1 (85)				
DESCRIPTION:	change duty durations, infeasible				
DUTY TYPE:	FLY	SUP	AFIT	PME	
Duration (years)	4	4	1	1	
Tolerance + (%)	10%	10%	5%	0%	
Tolerance - (%)	10%	10%	5%	0%	
GATE:	Gate 0	Gate 1	Gate 2	Gate 3	
Gate Miss Cost	4	3	2	1	
EXPERIENCE RQMT	Yes				
ATTRITION ADJUSTMENTS	Same as run # A1 (baseline values derived from input data)				
SANITY CHECKS	Pass				
# ITERATIONS	1730				
OBJ FUNCTION VALUE	7909				
FEASIBLE ?	No				
ARTIFICIALS: Number	30 variables				
Total artif flow	193.0				
Cost of artificials	7720				
OBJ $ - ARTIFICIAL $	189				
YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	999	796	831	783	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	273	467	444	328	293
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	0	N/A
Missed 2nd Gate	4	37	63	1	N/A
Missed 3rd Gate	33	27	30	79	N/A
TOTAL ASSIGNMENTS	1031	958	863	894	980
To FLY duties	543	630	517	593	553
To SUP duties	424	268	281	238	353
To AFIT duties	33	29	31	30	36
To PME duties	31	31	34	33	38
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	0	0	0	0	0
Will Miss GATE 1	0	0	0	0	0
Will Miss GATE 2	0	0	0	0	0
Will Miss GATE 3	16/26	14/0	30/18	72/0	3/10

RUN #	B2 (92)				
DESCRIPTION:	adjusted attrition, still infeasible				
DUTY TYPE:	FLY	SUP	AFIT	PME	
Duration (years)	4	4	1	1	
Tolerance + (%)	10%	10%	5%	0%	
Tolerance - (%)	10%	10%	5%	0%	
GATE:	Gate 0	Gate 1	Gate 2	Gate 3	
Gate Miss Cost	4	3	2	1	
EXPERIENCE RQMT	Yes				
ATTRITION ADJUSTMENTS	Large adjustments were made in an attempt to achieve feasibility. Total attrition was reduced by 187.				
SANITY CHECKS	Pass				
# ITERATIONS	1970				
OBJ FUNCTION VALUE	700				
FEASIBLE ?	No				
ARTIFICIALS: Number	2 variables				
Total artif flow	10.0				
Cost of artificials	200 (unit cost = 20)				
OBJ $ - ARTIFICIAL $	500				
YEAR	1	2	3	4	5
ROTATIONS INTO N'WORK	999	796	831	783	0
UFT and FAIP GAINS	317	315	315	299	299
TOTAL ATTRITION	263	399	419	300	237
ALREADY MISSED GATES					
Missed 1st Gate	0	0	0	0	N/A
Missed 2nd Gate	4	37	63	1	N/A
Missed 3rd Gate	33	27	30	79	N/A
TOTAL ASSIGNMENTS	1053	1015	972	1021	1161
To FLY duties	543	664	611	731	738
To SUP duties	446	291	296	225	354
To AFIT duties	33	29	31	32	36
To PME duties	31	31	34	33	33
ASSIGNMENTS WITH COST	(Number to Flying/Nonflying duties)				
Will Miss GATE 0	0	0	0	0	0
Will Miss GATE 1	0	0	0	0	0
Will Miss GATE 2	0/4	40/0	4/0	1/0	0
Will Miss GATE 3	17/36	14/4	119/19	167/0	2/14

Bibliography

1. Aronson, Jay E., T. E. Morton, and G. L. Thompson. "A Forward Simplex Method for Staircase Linear Programs," Management Science, 31: 664-679 (1985).

2. Aronson, Jay E., and G. L. Thompson. "The Solution of Multiperiod Personnel Planning Problems by the Forward Simplex Method," Large Scale Systems, 9: 129-139 (1985).

3. Charpie, Kenneth E., Jr. An Analysis of Chronic Personnel Shortages in the B-52 Radar Navigator Career Field. MS thesis, AFIT/GST/ENS/87M-4. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, March 1987.

4. Clark, Thomas D., Jr., and Kevin Lawson. "The Rotation and Assignment of Imbalanced Air Force Specialties: A Policy Analysis Model," Simulation, 43: 41-48 (July 1984).

5. Davies, G. S. "A Note on Continuous-Time Markov Manpower Models," Journal of Applied Probability, 22: 932-938 (1985).

6. Department of the Air Force. Air Force Policy Letter for Commanders. AFRP 190-1 5-87. Washington DC: HQ USAF, 1 May 1987.

7. Department of the Air Force. Flying: Flight Management. AFR 60-1. Washington DC: HQ USAF, 28 May 1985.

8. Department of the Air Force. Officer Personnel: Officer Assignments. AFR 36-20. Washington DC: HQ USAF, 25 October 1985.

9. Department of the Air Force. Rated Management Document, Volume I. Washington DC: HQ USAF, 23 April 1987.

10. Department of the Air Force. Rated Management Document, Volume II. Washington DC: HQ USAF, 23 April 1987.

11. Department of the Air Force. Schools: USAF Officer Professional Military Education System. AFR 53-8. Washington DC: HQ USAF, 24 October 1986.

12. Forrester, Jay W. "System Dynamics--Future Opportunities," TIMS Studies in the Management Sciences, Volume 14, edited by A. A. Legasto, Jr., J. W. Forrester, and J. M. Lyneis. New York: North-Holland Publishing Company, 1980.

13. Forrester, Jay W., and Peter M. Senge. "Tests for Building Confidence in System Dynamics Models," TIMS Studies in the Management Sciences, Volume 14, edited by A. A. Legasto, Jr., J. W. Forrester, and J. M. Lyneis. New York: North-Holland Publishing Company, 1980.

14. Hillier, Frederick S., and Gerald J. Lieberman. Introduction to Operations Research (Fourth Edition). Oakland, CA: Holden-Day, 1986.

15. Hooker, J. N. "Karmarkar's Linear Programming Algorithm," Interfaces, 16: 4, 75-90 (July-August 1986).

16. Jensen, Paul A., and J. Wesley Barnes. Network Flow Programming. Malabar, FL: Robert E. Krieger Publishing Co., 1987.

17. Jogerst, John D. Background Paper on Gate Supportable Inventory Methodology. Unpublished paper. Air Force Military Personnel Center, AFMPC/DPMYAF, Randolph AFB, TX, 27 October 1986.

18. Jogerst, John D. Talking Paper on Flying Gate Management. Unpublished paper. Air Force Military Personnel Center, AFMPC/DPMYAF, Randolph AFB, TX, 27 October 1986.

19. Kennington, J. L., and R. V. Helgason. Algorithms for Network Programming. New York: John Wiley and Sons, 1980.

20. Kennington, Jeffery L., and Alan Whisman. NETSID User's Guide. Technical Report 86-OR-01. Department of Operations Research, Southern Methodist University, Dallas, TX, June 1987.

21. Klingman, Darwin, Melissa Mead, and Nancy V. Phillips. "Network Optimization Models for Military Manpower Planning," Operational Research '84: Proceedings of the Tenth IFORS International Conference of Operational Research, Washington, D.C., U.S.A., August 6-10, 1984: 786-800. Amsterdam: North-Holland Publishing Company, 1984.

22. Knight, Jon M. The Aggregate Pilot Pipeline Model. Technical report, AFIT TR 78-6. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, November 1978 (AD-A064053).

23. Lawson, Kevin L. The Rotation/Assignment System of Imbalanced Air Force Specialty Codes Within Air Force Communications Command: A System Dynamics Model and Analysis. MS thesis, AFIT/GOR/OS/82D-8. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1982 (AD-A124779).

24. Liang, Timothy T., and Ben B. Buclatin. Improving the Utilization of Training Resources through Optimal Personnel Assignment in the U.S. Navy. Unpublished manuscript (accepted for publication by The European Journal of Operational Research). Navy Personnel Research and Development Center, San Diego, CA, April 1987.

25. McAlear, Kenneth E. The Aviation Career Incentive Act of 1974: An Analysis of Short-Range Results in the United States Air Force, 1974-1977. MS thesis. Army Command and General Staff College, Fort Leavenworth, KS, 9 June 1978 (AD-A058335).

26. Mehlmann, Alexander. "Markovian Manpower Models in Continuous Time," Journal of Applied Probability, 14: 249-259 (1977).

27. Price, W. L., and Marc Gravel. "Solving Network Manpower Problems With Side Constraints," European Journal of Operational Research, 15: 196-202 (February 1984).

28. Siverd, Samuel J., and Gerald L. Thompson. Ratio Goal Programming for Personnel Assignments. Contract N00014-75-C-0621. Carnegie-Melon University, Pittsburgh, PA, September 1979 (AD-A075220).

29. Sprague, Ralph H., Jr., and Eric D. Carlson. Building Effective Decision Support Systems. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1982.

30. Steuer, Ralph E. Multiple Criteria Optimization: Theory, Computation, and Application. New York: John Wiley and Sons, Inc., 1986.

31. Sutter, Brian L., Operations Research Analyst. Telephone interview. Analysis Division, Air Force Military Personnel Center, Randolph AFB, TX, 21 May 1987.

32. Sutter, Brian L., Operations Research Analyst. Telephone interview. Analysis Division, Air Force Military Personnel Center, Randolph AFB, TX, 26 May 1987.

33. Sutter, Brian L., Operations Research Analyst. Telephone interview. Analysis Division, Air Force Military Personnel Center, Randolph AFB, TX, O/A 15 June 1987.

34. Sutter, Brian L., Operations Research Analyst. Telephone interview. Analysis Division, Air Force Military Personnel Center, Randolph AFB, TX, 11 September 1987.

35. Sutter, Brian L., Operations Research Analyst. Telephone interview. Analysis Division, Air Force Military Personnel Center, Randolph AFB, TX, 6 November 1987.

36. Thompson, Gerald L. A Network Transshipment Model for Manpower Planning and Design. Contract N00014-76-C-0932. Carnegie-Melon University, Pittsburgh, PA, April 1978 (AD-A064345).

37. Wu, Nesa, and Richard Coppins. Linear Programming and Extensions. New York: McGraw-Hill, Inc., 1981.

Vita

Major Mark S. Olson was born on 12 October 1954 in Minneapolis, Minnesota. He graduated from Memorial High School in Eau Claire, Wisconsin, in 1972 and attended the United States Air Force Academy. He earned the degree of Bachelor of Science in Mathematics and was commissioned in June 1976. He earned his pilot's wings at Reese AFB, Texas, in 1977. He subsequently served as an F-4D aircraft commander with the 23rd Tactical Fighter Squadron at Spangdahlem Air Base, Germany. His next duty assignment was as a T-37 Instructor Pilot and Check Pilot at Williams AFB, Arizona. He then returned to the tactical forces as an A-10 Aircraft Commander, Flight Commander, and Instructor Pilot with the 74th TFS Flying Tigers at England AFB, Louisiana. In his off-duty time, he earned the degree of Master of Arts in Management from Webster University in 1985. He entered the Operations Research graduate degree program at the School of Engineering, Air Force Institute of Technology, in May 1986. He is married to the former Terri Ann Smith of Denver, Colorado, with whom he shares two children, Victoria and Michael.

Permanent address: 1201 South Clayton Street
Denver, Colorado 80210

8c. ADDRESS *(City, State, and ZIP Code)*	10. SOURCE OF FUNDING NUMBERS			
	PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO	WORK UNIT ACCESSION NO.

11. TITLE *(Include Security Classification)*

A NETWORK APPROACH TO RATED OFFICER GATE MANAGEMENT

12. PERSONAL AUTHOR(S)
Mark S. Olson, Major, USAF

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(Year, Month, Day)*	15. PAGE COUNT
MS Thesis	FROM ______ TO ______	1987 / December	232

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
FIELD	GROUP	SUB-GROUP	Network Flows, Mathematical Programming,
12	04		Military Personnel, Aviation Personnel,
15	01		Personnel Management

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

Thesis Committee Chairman: William F. Rowell, Lt Col, USAF
Assistant Professor

Approved for public release: IAW AFR 190-1.
LYNN E. WOLAVER 24 Feb 88
Dean for Re[illegible] Development
Air Force Ins[illegible]
Wright-P[illegible]

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	UNCLASSIFIED

The purpose of this study was to provide Air Force rated officer managers at the Air Force Military Personnel Center with a decision aid for the management of rated officer flying gates. Air Force rated officers are those officers who hold an aeronautical rating and are authorized to perform duties as pilots or navigators. Flying gates are milestones that must be achieved at certain phase points of a rated officer's career.

This study resulted in development of a single commodity network flow model with side constraints. This model is designed to represent the rotation of rated officers between flying and nonflying duties and provides a means for measuring overall attainment of flying gates. It is an aggregate model which provides general assignment guidance aimed at minimizing nonachievement of flying gate requirements, while maintaining required manning levels in flying and nonflying duties.

Initial analysis of model outputs indicates that the model solution may provide an avenue to improved gate management. Shortcomings of the model that bear further study include the level of detail provided by the model and the method used to model attrition of the rated officer force.

END
DATE
FILMED
3-88
DTIC